ADAPTIVE AND RECONFIGURABLE FLIGHT CONTROL

DISSERTATION

Yih-Shiun Huang, Lt. Col., ROCAF

AFIT/DS/ENG/01-02

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government.

AFIT/DS/ENG/01-02

Adaptive and Reconfigurable Flight Control

DISSERTATION

Presented to the Faculty

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

Yih-Shiun Huang, B.S., M.S.

Lieutenant Colonel, ROCAF

March 2001

AFIT/DS/ENG/01-02

Adaptive and Reconfigurable Flight Control

Yih-Shiun Huang, B.S., M.S.

Lieutenant Colonel, ROCAF

Approved:

M. Pachter — 6 March 2001

Dr. Meir Pachter
Committee Chairman

Date

Lt. Col. David R. Jacques — 6 MAR 01

Dr. Peter S. Maybeck — 6 March 2001

Dr. Peter S. Maybeck
Committee Member

Date

Mark E. Oxley — 6 March 2001

Dr. Mark E. Oxley
Committee Member

Date

Accepted:

7 Mar 01

Robert A. Calico, Jr.
Dean, Graduate School of Engineering and Management

Date

Acknowledgments

I would like to thank my advisor, Dr. Meir Pachter, Professor of Electrical Engineering. His frequent and patient assistance, encouragement and suggestions allowed me to make steady progress throughout the research phase. I also extend my gratitude to my committee members Dr. Peter S. Maybeck and Dr. Mark E. Oxley for their assistance, cooperation and comments.

I extend my special thanks to Mrs. Annette Robb, Director of International Affairs, and all the personnel in the IMSO for their assistance and help. I would like also to thank Mr. Vic Grazier who helped me and the other international students and our families since we arrived here.

Finally and most importantly, I want to thank my wife Mei-Yun, my sons Danny and Kevin for their patience and continuous support to overcome the difficulties of these years.

Yih-Shiun Huang

Table of Contents

List of Figures

List of Tables

Abstract

An indirect adaptive and reconfigurable flight control system is developed. The three-module controller consists of 1) a system identification module, 2) a parameter estimate smoother, and 3) a proportional and integral compensator for tracking control. Specifically: 1) The identification of a linear discrete-time control system's open-loop gain is addressed. The classical Kalman filter theory for linear control systems is extended and the control system's state and loop gain are jointly estimated on-line. Explicit formulae for the loop gain's estimate and estimation error covariance are derived. The estimate is unbiased and the predicted covariance is reliable. 2) An adaptive smoother is developed to reduce the fluctuations automatically in the gain estimate, and bursting, caused by instances of poor excitation. 3) Special attention is given to the design of a proportional and integral tracking controller. The outputs of the system identification and gain smoother modules are used to adjust the tracking controller's gain continuously in order to compensate for a possible reduction in the loop gain due to control surface area loss, thus achieving the benefits of adaptive and reconfigurable control. The performance of the adaptive and reconfigurable controller in the face of a simulated control surface failure is examined in carefully designed experiments. The adaptive controller developed in this dissertation and illustrated in a flight control context is applicable to a wide range of control problems.

Adaptive and Reconfigurable Flight Control

I. Introduction

Feedback is used to address the deleterious effects brought about by the unstructured environment in which the controlled plant is operating. This includes plant parameter uncertainty, unmodeled dynamics and input disturbances. At the same time, the benefits of feedback control are limited by actuator saturation and sensor noise, in particular when high gain feedback control is used [11].

System identification fits well into the feedback control paradigm for it allows us to acquire the estimates of the plant's parameters from measurements on the system's inputs and outputs using algorithms and software, but without adding extra hardware, i.e., sensors or actuators [13]. Thus, the uncertainty is reduced and lower gain feedback might be used. Unfortunately, system identification, which entails the estimation of all the (linear) plant's parameters, resides in the realm of nonlinear filtering. Moreover, system identification for adaptive and reconfigurable flight control requires a) the accurate and reliable estimation of the aircraft's stability and control derivatives with on-line operation, b) accurate and reliable estimation at low SNR, c) the use of a small sample, and d) no human intervention.

In the statistical linear regression paradigm, static system identification of the parameters of the dynamical system affords reliable real-time operation [19]. Static system identification is also well suited to aircraft parameter estimation since measurements of aircraft states and state rates are readily available [3, 5, 6].

It is shown in this dissertation that, in linear control systems, and provided that the dynamics (A) matrix is known, the exclusive estimation of the critical parameters of the control (B) matrix only is also reducible to a problem in linear regression and

therefore is amenable to linear analysis. Hence, a rigorous, and unbiased real-time estimate of the parameters of the control matrix and a reliable predicted estimation error covariance can be obtained. The classical Kalman filter theory for linear control systems is extended and the control system's state and loop gain are jointly estimated. Explicit formulae for the loop gain's estimate and predicted estimation error covariance are obtained. In this dissertation, this approach is pursued. A simplified (single input) version of this problem is addressed and an algorithm for the estimation of a single-input flight control system's critical loop gain parameter is developed [26]. The inclusion of a "forgetting factor" into this basic algorithm, or the employment of a sliding window, will afford the real-time identification of a time-varying parameter - the plant's open-loop gain. Thus, adaptive and reconfigurable flight control is possible.

Although the estimation problem addressed in this dissertation is amenable to a rigorous analysis and solution, the end-to-end adaptive control problem is nonlinear due to the presence of an unknown plant parameter. Therefore, we do not have the benefit of a Separation Theorem. Hence, an adaptive smoother is developed in this dissertation to reduce the fluctuations automatically in the parameter estimate caused by measurement noise, and bursting, caused by instances of poor excitation, prior to the use of the parameter estimate in the controller. As we move from window to window, the fluctuations in the parameter estimate are further exacerbated by the presence of modeling error. Hence the importance of the parameter estimate smoother.

Furthermore, special attention is given to the design of a proportional and integral tracking controller. The outputs of the system identification and gain smoother module are used to adjust the tracking controller's gain continuously.

Thus, an indirect adaptive and reconfigurable control system is developed. The controller consists of three modules: 1) A system identification module, 2) a parameter estimate smoother, and 3) a proportional and integral compensator for tracking

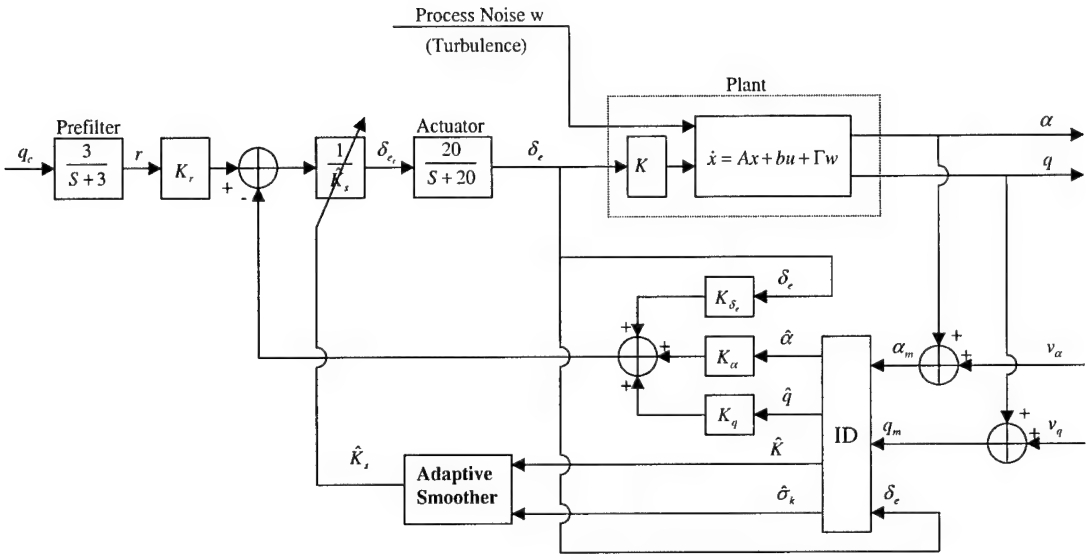


Figure 1. Adaptive and Reconfigurable Flight Control System

control. The architecture of an indirect adaptive and reconfigurable flight control system which incorporates the on-line loop gain identification algorithm, adaptive loop gain estimate smoother, and tracking controller developed in this dissertation is shown in Figure 1.

1.1 Assumptions

The plant dynamics are assumed to be known. We proceed under the assumption that the loss in elevator surface area does not significantly change the overall aircraft dynamics. For this reason, the identification of the aircraft's dynamics (A) matrix parameters is not undertaken and the focus is on the control matrix B, viz., the identification of the open-loop plant's gain, a reduction of which models the degree of failure, i.e., control surface area loss.

The system order is assumed known. Model order determination is not addressed in this dissertation for it is not a dominant issue in flight control [21]. The "short-period" aircraft dynamics approximation is used and the very low frequency

Phugoid dynamics can be neglected [18, 21]. This is so because the period of the Phugoid oscillation is about sixty seconds whereas the period of the short-period oscillation is about one second. Moreover, since this research deals with inner loop control and control surface failure, the control time horizon is short. This is to prevent aircraft departure, in particular when the open-loop plant is not stable and feedback control is used for stabilization. Thus, the time scale of interest for estimation and control is rather short, and neglecting the Phugoid is justified. The bandwidth of the actuator is sufficiently high and therefore the actuator and the aircraft's "short-period" dynamics are separated, and a first-order actuator model is satisfactory. These assumptions are validated in the simulation experiments, in which the unmodeled high frequency actuator, low frequency Phugoid dynamics, and parametric uncertainty, are included.

The aircraft model is assumed to be statically unstable to apply to modern fighter aircraft. Our three-module controller can also handle a stable aircraft plant, since the tracking controller is easier to design for a stable plant.

The benefit accrued from using a reduced order plant model for model based control design: When dealing with the short-period approximation of an aircraft, both the α (angle of attack) and q (pitch rate) signals are available for feedback, as is the elevator deflection δ_e, and thus full-state feedback control is possible - see, e.g., Figure 3 in the sequel.

1.2 System Identification Problem Statement

Indirect adaptive control relies on system identification. The plant truth model of the system identification algorithm is the linear discrete-time single-input multiple-

output stochastic control system (as shown in Figure 1)

$$
\begin{aligned}
x_{k+1} &= Ax_k + Kbu_k + \Gamma w_k, \quad E(w_k w_k^T) = Q, \quad x_k \in \Re^n && (1)\\
x_0 &= N(\overline{x}_{0,} P_{0_{xx}}) && (2)\\
K &= N(K_0, p_{0_{KK}}) && (3)\\
y_{k+1} &= Cx_{k+1} && (4)\\
z_{k+1} &= y_{k+1} + v_{k+1}, \quad E(v_{k+1} v_{k+1}^T) = R && (5)
\end{aligned}
$$

where

$$k = 0, 1, ..., N-1$$

In the special case of a single output, the matrix C is a row vector c^T and the measurement Eq. (5) is simplified to

$$z_{k+1} = y_{k+1} + v_{k+1}, \quad v_{k+1} = N(0, \sigma^2) \tag{6}$$

In the specific flight control application under consideration, the states α and q denote the aircraft's angle of attack and pitch rate, respectively, and the control signal δ_e is the elevator deflection.

The dynamics matrix A, the control vector b, the observation matrix C and the vector Γ are all known. The process noise intensity, Q, and the sensor noise intensity, R (or σ^2), are also known. In addition, the prior information specified in Eqs. (2) and (3) is provided. For an unfailed plant (aircraft) the loop gain $K = 1$ (by definition), until a failure at time t_f reduces the control derivative, and $K = K_1 < 1$ thereafter [3].

For a multiple-input system, the second term in the right-hand side of Eq. (1) becomes BKu_k, where B is the control matrix, K is the loop gain matrix and u_k is the input vector. In this dissertation we will deal with the single-input multiple-output stochastic control system only, since our aircraft plant is a single-input multiple-

output model. However, the methodology in this dissertation can be applied to a multiple-input multiple-output plant.

The objective is to identify the scalar loop gain K from the input sequence $u_0, u_1, ..., u_{N-1}$ and the recorded measurements, $z_1, z_2, ..., z_N$. A rigorous system identification algorithm is developed. The classical Kalman filter theory for linear control systems [4, 5, 6, 7] is extended and the control system's state and loop gain are jointly estimated. Explicit formula for the loop gain's estimate and predicted estimation error covariance are derived. The estimate of the system's state and the covariance of the state estimation error are also obtained. A sampling rate of 100Hz is used and in our estimation algorithm the continuous-time plant dynamics are discretized accordingly.

1.3 Adaptive and Reconfigurable Control System

The proposed three-module adaptive and reconfigurable control system is shown in Figure 1. The pilot inputs a pitch rate command, q_c, which is the reference signal for the adaptive controller. The commanded input is passed through a low pass prefilter and into a proportional (P) or proportional and integral (PI) controller designed to yield good tracking performance. The identification algorithm, i.e., the modified Kalman filter, is fed with the noise corrupted measurements of the states of the plant, α_m and q_m, and the input to the plant, δ_e. The $\widehat{\alpha}$, $\widehat{q}$ and $\widehat{K}$ estimates of α, q and K, respectively, are provided by the modified Kalman filter/system identification module at each time sample. The $\widehat{\alpha}$, $\widehat{q}$ and δ_e (elevator deflection angle) are fed back and summed with the reference signal r in the proportional and integral (linear) tracking compensator. The system identification module outputs an estimate $\widehat{K}$ of the loop gain and computes the loop gain estimation error variance. This information is fed to an adaptive smoother module which calculates a smoothed estimate $\widehat{K}_s$ of $\widehat{K}$. The filtered loop gain estimate ($\widehat{K}_s$) is fed back into the forward path after the summing junction of the states and signal r, but before the actuator,

to compensate for the changing open-loop gain K due to plant failure, i.e., a loss in control surface area. Thus, the control signal δ_{e_c} which is the input to the actuator is formed from the feedback filtered state measurements, the reference signal, and the output of the system identification and adaptive smoother modules. The indirect adaptive and reconfigurable control design methodology developed in this dissertation and illustrated in a flight control context is applicable to a wide range of control problems.

The plant model is representative of the longitudinal dynamics of an F-16 class aircraft. The open loop plant is unstable and feedback control is used for stabilization. The first order actuator model used herein is representative of the elevator of an F-16 class aircraft.

This dissertation is organized as follows. The basic concept of the system identification is presented in Chapter 2. The system identification algorithm for loop gain estimation is provided in Chapter 3. The adaptive loop gain smoothing algorithm is developed in Chapter 4 and the design of the linear tracking controller is discussed in Chapter 5. The aircraft model is discussed in Chapter 6. The experimental setup and the simulation results are presented in Chapter 7, followed by conclusions in Chapter 8.

II. System Identification

2.1 Introduction

This chapter provides general background knowledge of relevant and applicable system identification techniques. Both the classical and digital signal processing based method are explained and examples are given. Also, some concerns and design considerations related to system identification are addressed.

When modeling a system, there are two approaches to be considered, the deductive and empirical approaches [22]. With deductive modeling, the laws and equations found in physics and engineering are used to derive the proper model. With the empirical approach, i.e., statistical system identification, least squares and Kalman filter estimates are used to form a model for predicting the dynamics of the system.

System identification has more applications than just for modeling purposes. It is also a useful technique for model order reduction of a plant or compensator, for measuring or estimating parameters, and for real time adaptive control.

One of the challenges that needs to be addressed is modeling error, especially when the real order n of a best model (by some specified criterion) for the system is not known. Additional error sources are disturbances such as process and measurement noises. When determining the unknown parameters of a higher order model, the excess parameters of the physical system should equal zero. However, this does not happen because of noise. Rather, the critical parameter estimates are biased. Over-modeling is not advisable in system identification. Although it may seem logical to over-model a system, the extra parameters would be redundant and would make the determination of the critical system's parameters impossible. Therefore, it is better to under-estimate the order of the unknown system than to over-estimate it. Under-modeling requires determining values for the parameters that yield the best fit over a desired bandwidth.

2.2 Frequency Domain-Based Continuous-Time System Identification

Discrete-time system identification is well suited for a digital signal processing-based algorithm. Special attention needs to be given to continuous-time system identification.

Consider the continuous-time transfer function [22]

$$H(s) = \frac{y(s)}{u(s)} = \frac{b_1 s^{n-1} + b_2 s^{n-2} + ... + b_{n-1}s + b_n}{s^n - a_1 s^{n-1} - a_2 s^{n-2} - .... - a_{n-1}s - a_n} \ . \tag{7}$$

If we chose the input to be

$$u_k(t) = \sin \omega_k t, \ 0 \leq t \tag{8}$$

then the output is the phasor

$$\begin{aligned} y_k(t) &= (A_k + jB_k) \sin \omega_k t \\ &= \sqrt{A_k^2 + B_k^2} \sin(\omega_k t + \phi) \\ &= A \sin(\omega_k t + \phi) \end{aligned} \tag{9}$$

where

$$\phi = Arc \tan(\frac{B_k}{A_k}) \ . \tag{10}$$

Letting $s = j\omega$ and substitution into $H(s)$

$$\begin{aligned} H(s) &= \frac{y(s)}{u(s)} = A_k + jB_k \\ &= \frac{b_1(j\omega_k)^{n-1} + b_2(j\omega_k)^{n-2} + ... + b_{n-1}j\omega_k + b_n}{(j\omega_k)^n - a_1(j\omega_k)^{n-1} - a_2(j\omega_k)^{n-2} - .... - a_{n-1}(j\omega_k) - a_n}. \end{aligned} \tag{11}$$

Cross-multiplying and reducing creates a linear equation from which the unknown coefficients a_i and b_i can be obtained.

$$A_k + jB_k = \sum_{i=1}^{n}(A_k + jB_k)(j\omega_k)^{-i}a_i + \sum_{i=1}^{n}(j\omega_k)^{-i}b_i \tag{12}$$

Expanding the equation gives

$$\begin{aligned} A_k + jB_k &= \frac{1}{j}(A_k + jB_k)(\frac{1}{\omega_k}a_1 - \frac{1}{\omega_k^3}a_3 + \frac{1}{\omega_k^5}a_5 - \frac{1}{\omega_k^7}a_7 + ...) \\ &+\frac{1}{j}(\frac{1}{\omega_k}b_1 - \frac{1}{\omega_k^3}b_3 + \frac{1}{\omega_k^5}b_5 - \frac{1}{\omega_k^7}b_7 + ...) \\ &+(A_k + jB_k)(-\frac{1}{\omega_k^2}a_2 + \frac{1}{\omega_k^4}a_4 - \frac{1}{\omega_k^6}a_6 + ...) \\ &+(-\frac{1}{\omega_k^2}b_2 + \frac{1}{\omega_k^4}b_4 - \frac{1}{\omega_k^6}b_6 + ...). \end{aligned} \tag{13}$$

Multiplying both sides by j and expanding, we are able to match the real and complex portions of the right hand side the equation with the left. This will allow us to solve for the unknown coefficients:

$$\begin{aligned} -B_k &= A_k(\frac{1}{\omega_k}a_1 - \frac{1}{\omega_k^3}a_3 + \frac{1}{\omega_k^5}a_5 - \frac{1}{\omega_k^7}a_7 + ...) \\ &+B_k(\frac{1}{\omega_k^2}a_2 - \frac{1}{\omega_k^4}a_4 + \frac{1}{\omega_k^6}a_6 - ...) \\ &+(\frac{1}{\omega_k}b_1 - \frac{1}{\omega_k^3}b_3 + \frac{1}{\omega_k^5}b_5 - \frac{1}{\omega_k^7}b_7 + ...) \\ A_k &= B_k(\frac{1}{\omega_k}a_1 - \frac{1}{\omega_k^3}a_3 + \frac{1}{\omega_k^5}a_5 - \frac{1}{\omega_k^7}a_7 + ...) \\ &+A_k(-\frac{1}{\omega_k^2}a_2 + \frac{1}{\omega_k^4}a_4 - \frac{1}{\omega_k^6}a_6 + ...) \\ &+(-\frac{1}{\omega_k^2}b_2 + \frac{1}{\omega_k^4}b_4 - \frac{1}{\omega_k^6}b_6 + ...). \end{aligned} \tag{14}$$

For an n^{th} order SISO system, the control system is specified by $2n$ parameters. For this reason, we chose n sinusoidal test functions, each one producing 2 equations.

2.2.1 Second Order Example. Given the transfer function [22]

$$H(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}.$$

Knowing that we can measure the output phasor, the unknown parameters, ζ and ω_n, can be solved using the technique described in Eqs. (11)-(14):

$$A + jB = \frac{\omega_n^2}{-\omega^2 + 2\zeta\omega_n j\omega + \omega_n^2}.$$

Cross-multiplying and simplifying yields

$$(A + jB)(\omega_n^2 - \omega^2 + 2\zeta\omega_n j\omega) = \omega_n^2$$

which implies

$$(A-1)\omega_n^2 - 2B\omega\zeta\omega_n - A\omega^2 + j(B\omega_n^2 + 2A\omega\zeta\omega_n - B\omega^2) = 0.$$

Separating the imaginary and real terms yields

$$\begin{aligned} (A-1)\omega_n^2 - 2B\omega\zeta\omega_n &= A\omega^2 \\ B\omega_n^2 + 2A\omega\zeta\omega_n &= B\omega^2. \end{aligned}$$

Solving for natural frequency and damping ratio, the unknown parameters, yields

$$\begin{aligned} \omega_n &= \sqrt{\frac{A^2 + B^2}{A^2 + B^2 - A}}\omega \\ \zeta &= -\frac{B}{2\sqrt{(A^2 + B^2 - A)(A^2 + B^2)}}. \end{aligned}$$

Recall, A and B are coefficients of the output phasor and can be directly measured.

2.3 Stochastic Analysis

A goal of system identification is to find a "best fit" model of the system being analyzed [19]. The stochastic analysis acknowledges the noise in the measured input and output data. Using a least-squares approach, the sum of the squared errors between the true output of the model and the measured output is minimized. Using the least-squares approach and stochastic modeling, large errors and poor measurements are given less weight than more accurate measurements. The use of digital signal processing requires that our once-continuous system is now discretized.

In the case of a dynamic system, a system output can be described as [8, 19, 22]

$$y_{k+1} = a_1 y_k + a_2 y_{k-1} + ... + a_n y_{k-n+1} + b_1 u_k + b_2 u_{k-1} + ... + b_m u_{k-m+1} \tag{15}$$

and the measurement equation as

$$z_{k+1} = y_{k+1} + v_{k+1}, \qquad v_{k+1} \in N(0, \sigma^2) \tag{16}$$

or

$$y_k = z_k - v_k \tag{17}$$

which may contain process and disturbance noise. Therefore,

$$\begin{aligned} z_{k+1} - v_{k+1} \;=\; & a_1(z_k - v_k) + a_2(z_{k-1} - v_{k-1}) + ... + a_n(z_{k-n+1} - v_{k-n+1}) \\ & + b_1 u_k + b_2 u_{k-1} + ... + b_m u_{k-m+1}. \end{aligned} \tag{18}$$

By defining a noise vector, $\widetilde{V}_{k+1}$, we can rearrange the above equation as

$$z_{k+1} = a_1 z_k + a_2 z_{k-1} + ... + a_n z_{k-n+1} + b_1 u_k + b_2 u_{k-1} + ... + b_m u_{k-m+1} + \widetilde{V}_{k+1} \tag{19}$$

where

$$\widetilde{V}_{k+1} \triangleq v_{k+1} - a_1 v_k - a_2 v_{k-1} - .... - a_n v_{k-n+1}. \tag{20}$$

Now, the measurement equation is defined as

$$Z = H\Theta + \widetilde{V}, \quad E(\widetilde{V}\widetilde{V}^T) = R \tag{21}$$

where

$$Z \triangleq \begin{pmatrix} z_{k+1} \\ z_{k+2} \\ \vdots \\ z_{k+N+1} \end{pmatrix}_{(N+1)\times 1}, \quad \Theta \triangleq \begin{pmatrix} a_1 \\ \vdots \\ a_n \\ b_1 \\ \vdots \\ b_m \end{pmatrix}_{(m+n)\times 1}, \quad \widetilde{V} \triangleq \begin{pmatrix} \widetilde{V}_{k+1} \\ \widetilde{V}_{k+2} \\ \vdots \\ \widetilde{V}_{k+N+1} \end{pmatrix}_{(N+1)\times 1} \tag{22}$$

and

$$H \triangleq \begin{pmatrix} z_k & z_{k-1} & \cdots & z_{k-n+1} & u_k & u_{k-1} & \cdots & u_{k-m+1} \\ z_{k+1} & z_k & \cdots & z_{k-n+2} & u_{k+1} & u_k & \cdots & u_{k-m+2} \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{k+N} & z_{k+N-1} & \cdots & z_{k+N-n+1} & u_{k+N} & u_{k+N-1} & \cdots & u_{k-m+N} \end{pmatrix}_{(N+1)\times(m+n)} . \tag{23}$$

In this notation, m is the number of measurements and n is the number of parameters to be identified. We can now find the Least Squares estimate, $\widehat{\Theta}$, and the estimation error covariance matrix P [2, 22].

The Least Square estimate is

$$\widehat{\Theta} = (H^T H)^{-1} H^T Z \tag{24}$$

and the parameter estimation error covariance is

$$P = \sigma^2 (H^T H)^{-1}. \tag{25}$$

So far, the measurement and/or process noise has not been taken into account properly. A more rigorous method would be to solve for the parameter estimates using a weighted pseudo-inverse to account for noise in the system. The minimum-variance estimate is [4, 5]:

$$\widehat{\Theta}_{MV} = (H^T R^{-1} H)^{-1} H^T R^{-1} Z \tag{26}$$

and the minimum–variance estimation error is

$$P_{mv} = (H^T R^{-1} H)^{-1} \tag{27}$$

where R is the weighting matrix.

Eq. (24) calculates the parameter estimates using an unweighted pseudo-inverse. This is possible when the weighting matrix, R, is assumed to be a scaled unity matrix, viz., when cross-correlation does not exist and all scalar noises have the same variance. When this assumption is made, $R = \sigma^2 I$, where I is the identity matrix and σ^2 the variance of the noise, drops out of the parameter estimate equation and only σ^2 enters the calculated parameter estimation error covariance matrix.

2.4 Calculating R

R is affected by the noise created by our sensors when measuring the output of the system. Many times, R is represented as a diagonal matrix with σ^2 along the diagonal and zeros in the off-diagonal spaces, which implies no cross-correlation. This is often deemed adequate, knowing that each output has its own independent sensor to take measurements. In truth, the measurement noises are correlated and the sensors are coupled to some extent [22], and this causes off-diagonal terms in the R matrix. Earlier, in Eq. (5), we described R as being the expected value of the measurement noise times the transpose of that noise. It is shown from Eq. (22) that

the measurement noise is actually a noise vector, $\widetilde{V}$. This noise vector now produces off-diagonal terms in the R matrix.

The matrix R, using a two parameter estimation problem example, is now defined as

$$
\begin{aligned}
R &= E(\widetilde{V}\widetilde{V}^T) = E\left(\begin{pmatrix} \widetilde{V}_{k+1} \\ \vdots \\ \widetilde{V}_{k+N} \end{pmatrix} \begin{pmatrix} \widetilde{V}_{k+1} & \cdots & \widetilde{V}_{k+N} \end{pmatrix} \right) \\
&= E\left(\begin{pmatrix} v_{k+1} - a_1 v_k \\ v_{k+2} - a_1 v_{k+1} \\ \vdots \\ v_{k+N} - a_1 v_{k+N-1} \end{pmatrix} \right. \\
&\qquad \left. \begin{pmatrix} v_{k+1} - a_1 v_k & v_{k+2} - a_1 v_{k+1} & \vdots & v_{k+N} - a_1 v_{k+N-1} \end{pmatrix} \right) \\
&= E\begin{pmatrix} (v_{k+1} - a_1 v_k)(v_{k+1} - a_1 v_k) & (v_{k+1} - a_1 v_k)(v_{k+2} - a_1 v_{k+1}) & \cdots \\ (v_{k+2} - a_1 v_{k+1})(v_{k+1} - a_1 v_k) & (v_{k+2} - a_1 v_{k+1})(v_{k+2} - a_1 v_{k+1}) & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix} \\
&= \begin{pmatrix} \sigma^2 + a^2\sigma^2 & -a\sigma^2 & 0 & 0 & \cdots \\ -a\sigma^2 & \sigma^2 + a^2\sigma^2 & -a\sigma^2 & 0 & \cdots \\ 0 & -a\sigma^2 & \sigma^2 + a^2\sigma^2 & -a\sigma^2 & 0 \\ 0 & 0 & -a\sigma^2 & \sigma^2 + a^2\sigma^2 & \cdots \\ \vdots & \vdots & 0 & -a\sigma^2 & \ddots \end{pmatrix}. \qquad (28)
\end{aligned}
$$

Finally,

$$R = \sigma^2 \begin{pmatrix} 1+a^2 & -a & 0 & \cdots \\ -a & 1+a^2 & -a & \cdots \\ 0 & -a & 1+a^2 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{pmatrix} \tag{29}$$

$$= \sigma^2 \widetilde{R}. \tag{30}$$

Now, Eqs. (26) and (27) become

$$\widehat{\Theta}_{MV} = (H^T \widetilde{R}^{-1} H)^{-1} H^T \widetilde{R}^{-1} Z \tag{31}$$

$$P_{mv} = \sigma^2 (H^T \widetilde{R}^{-1} H)^{-1}. \tag{32}$$

Note that the intensity of noise, σ, does not affect the parameter estimate $\widehat{\Theta}_{MV}$.

In summary, the parameters of an unknown dynamic system can be identified by using a sinusoidal test function as an input to that unknown plant and observing the phasor output. However, care must be taken not to over-model the unknown plant. When noise is included in the system dynamics, a stochastic analysis of the measurement situation must be performed to account properly for the effects of noise. A discrete-time plant model was also discussed and the input and measurement equations, now with a noise term, were defined . It was then shown how the strength of the noise can be represented as a weighting matrix, R, and applied to the least squares formula, to provide more accurate parameter estimates.

III. Stochastic Modeling

A Kalman filter is a data processing algorithm that uses all available data, such as plant model, initial conditions, and statistical descriptions of any biases, measurement noise or process noise [15, 16]. This information is fed into the propagate/update algorithm which then optimally derives an estimated value for the system's state in a way that minimizes estimation error variance. The rigorous Kalman filtering paradigm for linear systems can be extended to provide an estimate of the control matrix B. Our main result is the following system identification algorithm.

3.1 System Identification Algorithm

Since digital signal processing is used, a discrete-time dynamical model considered in this work is used. The loop gain system identification algorithm is developed by Sillence [26] and is concisely presented in Theorem 1.

Theorem 1 *Consider the following linear estimation problem. The linear dynamical system is*

$$x_{k+1} = Ax_k + Kbu_k + \Gamma w_k, \quad E(w_k w_k^T) = Q, \quad k = 0, 1, ..., N-1 \tag{33}$$

the prior information is

$$x_0 \in N(\overline{x}_0, P_{0_x}) \tag{34}$$

$$K \in N(K_0, P_{0_K}) \tag{35}$$

the output signal

$$y_{k+1} = Cx_{k+1} \tag{36}$$

and the observation equation is

$$z_{k+1} = y_{k+1} + v_{k+1}, \quad E(v_{k+1}v_{k+1}^T) = R. \tag{37}$$

The matrices A, b, C and Γ are known. The respective Gaussian zero mean process noise and measurement noise covariance matrices, Q and R, are also known. The open-loop gain K is not known.

Denote by $\widehat{x}_k$ and $\widehat{K}_k$ the respective estimates of the state x_k and the loop gain K at time k, given the measurements record $z_1, ..., z_k$, the input sequence $u_0, ..., u_{k-1}$, and the prior information on x_0 and K. The covariance of the estimation error of the $\begin{pmatrix} x_k \\ K \end{pmatrix}$ vector is denoted by the partitioned matrix $P_k = \begin{pmatrix} P_{k_{xx}} & p_{k_{xK}} \\ p_{k_{xK}}^T & p_{k_{KK}} \end{pmatrix}$.

Initially, set

$$\widehat{x}_0 \triangleq \overline{x}_0, \quad \widehat{K}_0 \triangleq K_0, \quad P_{0_{xx}} \triangleq P_{0_x}, \quad p_{0_{KK}} \triangleq P_{0_K}, \quad p_{0_{xK}} \triangleq 0. \tag{38}$$

Then, for $k = 0, 1, ..., N-1$, the state and gain estimates are updated as follows

$$\widehat{x}_{k+1} = A\widehat{x}_k + \widehat{K}_k b u_k + K_x(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k) \tag{39}$$

$$\widehat{K}_{k+1} = \widehat{K}_k + K_K(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k) \tag{40}$$

where the Kalman gains

$$\begin{aligned} K_x = {} & \{AP_{k_{xx}}A^TC^T + u_k[Ap_{k_{xK}}(Cb)^T + b(CAp_{k_{xK}})^T] \\ & + u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q \Gamma^T C^T\} \times \{CAP_{k_{xx}}A^TC^T \\ & + u_k[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T] \\ & + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\}^{-1} \end{aligned} \tag{41}$$

and

$$\begin{aligned} K_K &= [(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T] \times \{CAP_{k_{xx}}A^TC^T \\ &\quad + u_k[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T] \\ &\quad + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\}^{-1}. \end{aligned} \tag{42}$$

Furthermore, the estimation error covariances are

$$\begin{aligned} P_{k+1_{xx}} &= \{[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T)] \\ &\quad + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T]^{-1} + C^T R^{-1} C\}^{-1} \end{aligned} \tag{43}$$

$$\begin{aligned} p_{k+1_{KK}} &= p_{k_{KK}} - [(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T] \times \{CAP_{k_{xx}}A^TC^T \\ &\quad + u_k[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T] \\ &\quad + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\}^{-1} \\ &\quad \times [(CAp_{k_{xK}}) + u_k p_{k_{KK}}(Cb)] \end{aligned} \tag{44}$$

$$\begin{aligned} p_{k+1_{xK}} &= Ap_{k_{xK}} + u_k p_{k_{KK}} b - \{AP_{k_{xx}}A^TC^T + u_k[Ap_{k_{xK}}(Cb)^T + b(CAp_{k_{xK}})^T] \\ &\quad + u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q\Gamma^T C^T\} \\ &\quad \times \{CAP_{k_{xx}}A^TC^T + u_k[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T] \\ &\quad + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\}^{-1} \\ &\quad \times [(CAp_{k_{xK}}) + u_k p_{k_{KK}}(Cb)]. \end{aligned} \tag{45}$$

□

3.2 Proof of Theorem 1

We shall require the complete Matrix Inversion Lemma (MIL)

Lemma 2 *Assume the relevant matrices are compatible and invertible. Then*

$$\left(A_1 - A_2 A_4^{-1} A_3\right)^{-1} = A_1^{-1} + A_1^{-1} A_2 \left(A_4 - A_3 A_1^{-1} A_2\right)^{-1} A_3 A_1^{-1}. \tag{46}$$

□

Since the unknown loop gain is a constant, we augment the dynamics as follows.

$$K_{k+1} = K_k. \tag{47}$$

Hence, the augmented state dynamics evolve in $\Re^{n+1}$ and are

$$\begin{pmatrix} x_{k+1} \\ K_{k+1} \end{pmatrix} = \begin{pmatrix} A & u_k b \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x_k \\ K_k \end{pmatrix} + \begin{pmatrix} \Gamma \\ 0 \end{pmatrix} w_k \tag{48}$$

and the measurement equation is

$$z_{k+1} = \begin{pmatrix} C & \vdots & 0 \end{pmatrix} \begin{pmatrix} x_{k+1} \\ K_{k+1} \end{pmatrix} + v_{k+1}. \tag{49}$$

As can be seen, the equations are similar to that of the deterministic model except for noise now being modeled into the system. Here, the w_k and v_{k+1} represent the process noise and measurement noise, respectively. The covariances of these noises are represented by Q and R in the stochastic model. The values of Q and R were defined in Chapter 1.

The prior information at time instant k is

$$\begin{pmatrix} x_k \\ K_k \end{pmatrix} \in N\left(\begin{pmatrix} \widehat{x}_k \\ \widehat{K}_k \end{pmatrix}, P_k\right) \tag{50}$$

where

$$P_k = \begin{pmatrix} P_{k_{xx}} & p_{k_{xK}} \\ p_{k_{xK}}^T & p_{k_{KK}} \end{pmatrix} \tag{51}$$

is the estimation error covariance matrix. The elements of P_k are

$$P_{k_{xx}} \in \Re^{n \times n}, \quad p_{k_{xK}} \in \Re^n, \quad p_{k_{KK}} \in \Re^1. \tag{52}$$

Hence, before the z_{k+1} measurement is recorded, the augmented state

$$\begin{aligned} \begin{pmatrix} x_{k+1} \\ K_{k+1} \end{pmatrix} \quad &\in \quad N\left(\begin{pmatrix} A & u_k b \\ 0 & 1 \end{pmatrix}\begin{pmatrix} \widehat{x}_k \\ \widehat{K}_k \end{pmatrix}, \right. \\ &\left. \begin{pmatrix} A & u_k b \\ 0 & 1 \end{pmatrix} P_k \begin{pmatrix} A^T & 0 \\ u_k b^T & 1 \end{pmatrix} + \begin{pmatrix} \Gamma Q \Gamma^T & 0 \\ 0 & 0 \end{pmatrix}\right) \\ &= \quad N\left(\begin{pmatrix} A\widehat{x}_k + \widehat{K}_k b u_k \\ \widehat{K}_k \end{pmatrix}, \right. \\ &\left. \begin{pmatrix} AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + & \vdots & Ap_{k_{xK}} + u_k p_{k_{KK}} b \\ bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T & \vdots & \\ \cdots\cdots\cdots\cdots\cdots\cdots & \vdots & \cdots\cdots\cdots\cdots \\ p_{k_{xK}}^T A^T + u_k p_{k_{KK}} b^T & \vdots & p_{k_{KK}} \end{pmatrix}\right). \end{aligned} \tag{53}$$

Next, apply the Bayesian estimation formula and obtain

$$\widehat{x}_k^+ = \widehat{x}_k^- + K(z - H\widehat{x}_k) \tag{54}$$

$$\begin{pmatrix} \widehat{x}_{k+1} \\ \widehat{K}_{k+1} \end{pmatrix} = \begin{pmatrix} A\widehat{x}_k + \widehat{K}_k b u_k \\ \widehat{K}_k \end{pmatrix} + K \left(z_{k+1} - \begin{pmatrix} C & \vdots & 0 \end{pmatrix} \begin{pmatrix} A\widehat{x}_k + \widehat{K}_k b u_k \\ \widehat{K}_k \end{pmatrix} \right)$$

$$= \begin{pmatrix} A\widehat{x}_k + \widehat{K}_k b u_k \\ \widehat{K}_k \end{pmatrix} + K \left(z_{k+1} - CA\widehat{x}_k - u_k \widehat{K}_k C b \right) \tag{55}$$

where the Kalman gain

$$K = \begin{pmatrix} AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + & \vdots & Ap_{k_{xK}} + u_k p_{k_{KK}} b \\ bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T & \vdots & \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \vdots & \cdots\cdots\cdots\cdots \\ p_{k_{xK}}^T A^T + u_k p_{k_{KK}} b^T & \vdots & p_{k_{KK}} \end{pmatrix} \times$$

$$\begin{pmatrix} C^T \\ 0 \end{pmatrix} \times \left\{ CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T \right] \right.$$

$$\left. + u_k^2 p_{k_{KK}} (Cb)(Cb)^T + C\Gamma Q \Gamma^T C^T + R \right\}^{-1}$$

$$= \begin{pmatrix} AP_{k_{xx}}A^T C^T + u_k \left[Ap_{k_{xK}}(Cb)^T + b\left(CAp_{k_{xK}}\right)^T \right] \\ + u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q \Gamma^T C^T \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ (CAp_{k_{xK}})^T + u_k p_{k_{KK}} (Cb)^T \end{pmatrix} \times$$

$$\left\{ CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T \right] \right.$$

$$\left. + u_k^2 p_{k_{KK}} (Cb)(Cb)^T + C\Gamma Q \Gamma^T C^T + R \right\}^{-1}. \tag{56}$$

Finally,

$$P_{k+1_{(x,K)}} = P_{k_{(x,K)}} - K \begin{pmatrix} C & \vdots & 0 \end{pmatrix} P_{k_{(x,K)}}. \tag{57}$$

Hence, we calculate

$$
P_{k+1_{(x,K)}} = \begin{pmatrix} AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + & \vdots & Ap_{k_{xK}} + u_k p_{k_{KK}}b \\ bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T & \vdots & \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots & \vdots & \cdots\cdots\cdots \\ p_{k_{xK}}^T A^T + u_k p_{k_{KK}} b^T & \vdots & p_{k_{KK}} \end{pmatrix} -
$$

$$
\begin{pmatrix} \left\{AP_{k_{xx}}A^TC^T + u_k\left[Ap_{k_{xK}}(Cb)^T + b\left(CAp_{k_{xK}}\right)^T\right]\right. & \vdots \\ \left. +u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q \Gamma^T C^T\right\} & \vdots \\ \times\left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. & \vdots \\ \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} \times \left[CAp_{k_{xx}}A^T\right. & \vdots \\ \left. +u_k\left(CAp_{k_{xK}}b^T + CbAp_{k_{xK}}^T A^T\right) + u_k^2 p_{k_{KK}} Cbb^T + C\Gamma Q\Gamma^T\right] & \vdots \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots & \vdots \\ \left[(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T\right] & \vdots \\ \times\left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. & \vdots \\ \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} \times \left[CAP_{k_{xx}}A^T\right. & \vdots \\ \left. +u_k\left(CAp_{k_{xK}}b^T + CbAp_{k_{xK}}^T A^T\right) + u_k^2 p_{k_{KK}} Cbb^T + C\Gamma Q\Gamma^T\right] & \vdots \end{pmatrix}
$$

$$
\begin{pmatrix} \vdots & \left\{AP_{k_{xx}}A^TC^T + u_k\left[Ap_{k_{xK}}(Cb)^T + b\left(CAp_{k_{xK}}\right)^T\right]\right. \\ \vdots & \left. +u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q\Gamma^T C^T\right\} \\ \vdots & \times\left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\ \vdots & \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} \\ \vdots & \times\left(CAp_{k_{xK}} + u_k p_{k_{KK}} Cb\right) \\ \vdots & \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \vdots & \left[(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T\right] \\ \vdots & \times\left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\ \vdots & \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} \\ \vdots & \times\left(CAp_{k_{xK}} + u_k p_{k_{KK}} Cb\right) \end{pmatrix}. \tag{58}
$$

Thus,

$$\begin{aligned}
P_{k+1_{xx}} &= [AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T] \\
&\quad \times \Big\{ \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T\right]^{-1} \\
&\quad -C^T \left\{CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\
&\quad \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} C \Big\} \\
&\quad \times [AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T]. \qquad (59)
\end{aligned}$$

Next, apply the MIL (Lemma 2) to the expression in the outer curly brackets from Eq. (59), viz.,

$$\begin{aligned}
&\Big\{ \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T\right]^{-1} \\
&-C^T \left\{CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\
&\left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} C \Big\}^{-1}
\end{aligned}$$

where we set

$$\begin{aligned}
A_1 &= \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q\Gamma^T\right]^{-1} \\
A_2 &= C^T \\
A_3 &= C \\
A_4 &= \left\{CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\
&\quad \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}
\end{aligned}$$

we obtain

$$
\begin{aligned}
& \left\{ \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right]^{-1} \right. \\
& -C^T \left\{ CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T \right] \right. \\
& \left. \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q \Gamma^T C^T + R \right\}^{-1} C \right\}^{-1} \\
= \; & \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right] \\
& + \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right] C^T \\
& \times \left\{ \left\{ CAP_{k_{xx}}A^T C^T + u_k \left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T \right] \right. \right. \\
& \left. +u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q \Gamma^T C^T + R \right\} \\
& \left. -C \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right] C^T \right\}^{-1} \\
& \times C \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right].
\end{aligned}
$$

Reducing the above gives

$$
\begin{aligned}
& \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right] \\
& \times \left\{ \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right]^{-1} + C^T R^{-1} C \right\} \\
& \times \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right].
\end{aligned}
$$

Hence, Eq. (59) can now be reduced to

$$
\begin{aligned}
P_{k+1_{xx}} = & \left\{ \left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}} bb^T + \Gamma Q \Gamma^T \right]^{-1} \right. \\
& \left. +C^T R^{-1} C \right\}^{-1}. \qquad (60)
\end{aligned}
$$

In addition,

$$\begin{aligned} p_{k+1_{KK}} &= p_{k_{KK}} - \left[(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T\right] \\ &\quad \times \left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\ &\quad \left. + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1}(CAp_{k_{xK}} + u_k p_{k_{KK}}Cb) \end{aligned} \tag{61}$$

and

$$\begin{aligned} p_{k+1_{xK}} &= Ap_{k_{xK}} + u_k p_{k_{KK}} b - \left\{AP_{k_{xx}}A^TC^T + u_k\left[Ap_{k_{xK}}(Cb)^T + b(CAp_{k_{xK}})^T\right]\right. \\ &\quad \left. + u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q\Gamma^T C^T\right\} \\ &\quad \left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\ &\quad \left. + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1}(CAp_{k_{xK}} + u_k p_{k_{KK}}Cb)\,. \end{aligned} \tag{62}$$

We also partition the Kalman gain vector as follows

$$K = \begin{pmatrix} K_x \\ K_K \end{pmatrix} \tag{63}$$

where

$$\begin{aligned} K_x &= \left\{AP_{k_{xx}}A^TC^T + u_k\left[Ap_{k_{xK}}(Cb)^T + b(CAp_{k_{xK}})^T\right]\right. \\ &\quad \left. + u_k^2 p_{k_{KK}} b(Cb)^T + \Gamma Q\Gamma^T C^T\right\} \\ &\quad \times \left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\ &\quad \left. + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1} \end{aligned} \tag{64}$$

and

$$
\begin{aligned}
K_K &= \left[(CAp_{k_{xK}})^T + u_k p_{k_{KK}}(Cb)^T\right] \\
&\quad \times \left\{CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right]\right. \\
&\quad \left. + u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R\right\}^{-1}. \qquad (65)
\end{aligned}
$$

Hence, we finally obtain

$$
\begin{aligned}
\widehat{x}_{k+1} &= A\widehat{x}_k + \widehat{K}_k b u_k + K_x(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k) \qquad (66) \\
\widehat{K}_{k+1} &= \widehat{K}_k + K_K(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k). \qquad (67)
\end{aligned}
$$

□

Proposition 3 *An additional application of the MIL will reduce the number of matrix inversions such that only the low-order matrix*

$$
\begin{aligned}
&CAP_{k_{xx}}A^TC^T + u_k\left[CAp_{k_{xK}}(Cb)^T + (Cb)(CAp_{k_{xK}})^T\right] \\
&+u_k^2 p_{k_{KK}}(Cb)(Cb)^T + C\Gamma Q\Gamma^T C^T + R
\end{aligned}
$$

needs to be inverted.

□

3.3 Discussion

It is important to realize that the absence of complete plant information, viz., the uncertainty in the loop gain parameter K, causes both the parameter and the state estimation error covariances to be dependent on the input signal - see, e.g., the covariance equations (43)-(45) in Theorem 1. This is a major departure from the classical state estimation paradigm in linear control theory. Thus, the loop gain estimate $\widehat{K}$ (and also the loop gain estimation error covariance) are now control-

dependent. Obviously, the best loop gain estimate is obtained at the end of the estimation interval, at time N. In addition, the algorithm-provided loop gain and state estimates are correlated. Furthermore, the loop gain and state estimates' dependence on the input signal is nonlinear. The input signal dependence of the loop gain and state estimation error covariances, is a unique manifestation of the *dual control* effect. This means that the estimation error variance is dependent on the input signal, which is not the case in classical linear state estimation/Kalman filtering.

3.3.1 Example 1. Consider the classical Kalman Filter paradigm where the loop gain K is known, i.e., $K = 1$. In this special case

$$p_{0_{KK}} = 0, \quad p_{0_{xK}} = 0, \quad p_{k_{KK}} = 0, \quad p_{k_{xK}} = 0 \qquad \text{for all } k = 1, 2, 3, ...$$

and it follows that

$$\begin{aligned} P_k &= P_{k_{xx}} \\ K_K &= 0 \\ K_x &= \left(AP_{k_{xx}}A^T + \Gamma Q\Gamma^T\right) C_T \\ &\quad \times \left(CAP_{k_{xx}}A^TC^T + C\Gamma Q\Gamma^TC^T + R\right)^{-1} \\ P_{k+1_{xx}} &= \left[\left(AP_{k_{xx}}A^T + \Gamma Q\Gamma^T\right)^{-1} + C^TR^{-1}C\right]^{-1}. \end{aligned}$$

Thus, the classical Kalman filter formulae are recovered.

Remark 4 *If x_0 is known, viz., $x_0 \in N(\overline{x}_0, 0)$, i.e., $P_{0_x} = 0$, and only the loop gain parameter K is not known, i.e., $P_{0_{xx}} = 0$, $p_{0_{xK}} = 0$, one nevertheless has to deal with an uncertain x at time k (even if $\Gamma = 0$ and if there is no process noise) and one must propagate* $\begin{pmatrix} \widehat{x}_k \\ \widehat{K}_k \end{pmatrix}$ *and* $P_{k_{(n+1)\times(n+1)}}$.

□

3.3.2 Example 2. Special case: C is a row vector (i.e., a scalar measurement is being used), then the estimation algorithm is

$$\begin{aligned}\widehat{x}_{k+1} &= A\widehat{x}_k + \widehat{K}_k b u_k + K_x(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k) \\ \widehat{K}_{k+1} &= \widehat{K}_k + K_K(z_{k+1} - CA\widehat{x}_k - \widehat{K}_k C b u_k)\end{aligned}$$

where the Kalman gain for state estimation is

$$\begin{aligned}K_x &= \frac{1}{X}\left\{AP_{k_{xx}}A^T C^T + u_k\left[(Cb)Ap_{k_{xK}} + (CAp_{k_{xK}})b\right]\right. \\ &\quad \left. + u_k^2(Cb)p_{k_{KK}}b + \Gamma Q\Gamma^T C^T\right\}\end{aligned}$$

and where the scalar X is given by

$$\begin{aligned}X &\triangleq CAP_{k_{xx}}A^T C^T + 2u_k CbAp_{k_{xK}} \\ &\quad + u_k^2(Cb)p_{k_{KK}} + C\Gamma Q\Gamma^T C^T + R.\end{aligned}$$

The Kalman gain for loop gain estimation is

$$K_K = \frac{\left[CAp_{k_{xK}} + u_k CbAp_{k_{KK}}\right]}{X}.$$

Finally, the estimation error covariances are

$$\begin{aligned}P_{k+1_{xx}} &= \left\{\left[AP_{k_{xx}}A^T + u_k(Ap_{k_{xK}}b^T + bp_{k_{xK}}^T A^T) + u_k^2 p_{k_{KK}}bb^T + \Gamma Q\Gamma^T\right]^{-1}\right. \\ &\quad \left. + \frac{1}{R}C^T C\right\}^{-1} \\ p_{k+1_{KK}} &= p_{k_{KK}} - \frac{\left[CAp_{k_{xK}} + u_k CbAp_{k_{KK}}\right]^2}{X} \\ p_{k+1_{xK}} &= Ap_{k_{xK}} + u_k p_{k_{KK}}b - \frac{\left[CAp_{k_{xK}} + u_k CbAp_{k_{KK}}\right]}{X} \\ &\quad \times \left\{AP_{k_{xx}}A^T C^T + u_k\left[(Cb)Ap_{k_{xK}} + (CAp_{k_{xK}})b\right]\right. \\ &\quad \left. + u_k^2(Cb)p_{k_{KK}}b + \Gamma Q\Gamma^T C^T\right\}.\end{aligned}$$

In summary, a system identification algorithm was developed to identify a system's loop gain, K.

IV. Adaptive Parameter Smoothing

In conventional indirect adaptive control a two-module controller consisting of a compensator and a system identification module is used. The compensator is synthesized on-line using a model-based controller design methodology. The system identification module provides the plant parameter estimate. The latter is used in the on-line compensator synthesis algorithm to update the plant model and thus modify the compensator accordingly, which yields adaptive control action. There is a tendency to rely on *assumed certainty equivalence* and directly insert the parameter estimate into the compensator synthesis formula. This course of action is to a large extent motivated by the classical solution of the LQG problem in which an LQR state feedback compensator is used in tandem with a Kalman filter which provides the state estimate [17].

At this point it is worthwhile to recall the LQG paradigm momentarily: the plant is linear and known and the cost functional is quadratic in the state and control signals. The state estimation problem for linear control systems with known dynamics, control and observation (A, B, C) matrices resides in the realm of linear regression, and therefore the Kalman filter solution of the minimum variance state estimation problem yields an unbiased estimate of the state. An LQR compensator directly operates on the Kalman filter-provided state estimate, viz., the LQR compensator is used in tandem with the Kalman filter. The LQG controller, which consists of two modules, the Kalman filter and the fixed LQR compensator, is optimal.

Our plant, specified in Eqs. (1) - (5), is linear, however it contains the unknown parameter K which quantifies the degree of control power loss in the control effector. Thus, the state and the plant parameter K are to be jointly estimated. It is remarkable that also our system identification algorithm, as stated in *Theorem* 1, yields an unbiased state and parameter estimate and a reliable predicted estimate

error covariance. Moreover, in the special case where $p_{0_{KK}} = 0$ and $K_0 = 1$, viz., the parameter is known and therefore the (linear) plant is completely known, the classical Kalman filter state estimation formulae are recovered - see, e.g., *Example* 1 in Chapter 3. Indeed, our derivation of the system identification algorithm is rigorous and:

- The estimate is unbiased.
- The predicted covariance of the estimation error is reliable.

In addition, it is noteworthy that when the plant is completely known and one is exclusively interested in the state estimate, as is the case in the classical Kalman filtering paradigm, the quality of the state estimate, viz., the predicted covariance of the state estimation error, is not dependent on the plant input. Indeed, the covariance of the state estimation error is not used in the LQG controller. However, in the adaptive control case, where the plant parameter K is not known, the predicted state and parameter estimation error covariance are dependent on the plant input - see, e.g., Eqs. (43) - (45). Thus, in adaptive control, the quality of the state and parameter estimate is dependent on the input signal. Evidently, when the predicted estimation error covariance is small, the quality of the parameter estimate is good, and, conversely, when the predicted estimation error covariance is large, we cannot be certain about the true value of the parameter and the system identification algorithm-provided parameter estimate might be far from the true parameter. Thus, the control signal-dependence of the quality of the parameter (and state) estimate motivates one to refer to the excitation quality of the control signal. Good excitation yields a predicted parameter estimate error variance which is small, and thus is conducive to a well designed parameter estimation experiment.

The following is crucial. In our specific situation where the system identification algorithm is rigorously derived, an unbiased state and parameter estimate and a reliable computed (predicted) estimation error covariance are obtained. Hence,

the excitation quality is indeed directly reflected in the size of the predicted state and parameter estimation error covariance. In other words, we have a direct measurement of the, otherwise somewhat nebulous, degree of excitation - it is given by the size of the computed parameter estimation error variance, provided the latter is reliable, which, in our case, it is. Indeed, the importance of the computed parameter estimation error variance being reliable cannot be overestimated. When an erroneous computed estimation error variance is used in, e.g., state estimation, the state estimate becomes "biased" and one then refers to filter divergence. This is a common occurrence in extended Kalman filtering [16, 17]. Our work hinges on the computed parameter estimation error variance being reliable, by virtue of the rigorous system identification algorithm developed in Chapter 3.

Now, the dependence of the quality of the system identification algorithm-provided parameter and/or state estimate on the control signal is the root cause of the *dual control* effect observed in nonlinear stochastic control and in adaptive control [17]. When the straightforward *assumed certainty equivalence* principle is used in adaptive control, the *dual control* effect is responsible for the *bursting* phenomenon often observed in adaptive control [1, 17]. This then invalidates the applicability of assumed certainty equivalence outside the very circumscribed LQG paradigm, where the quality of the estimate is not dependent on the control signal and the separation principle upon which certainty equivalence hinges is rigorously proven.

4.1 Fixed-Weights Parameter Filter

Evidently, when the estimation error variance is large, due to low excitation, and in the absence of a *Separation Theorem* [1, 12, 25, 27], we are not encouraged to boldly insert the new plant parameter estimate into the compensator synthesis equation. Indeed, when the parameter estimation error variance is large, measurement noise will cause the parameter estimate to vary wildly from window to window, as

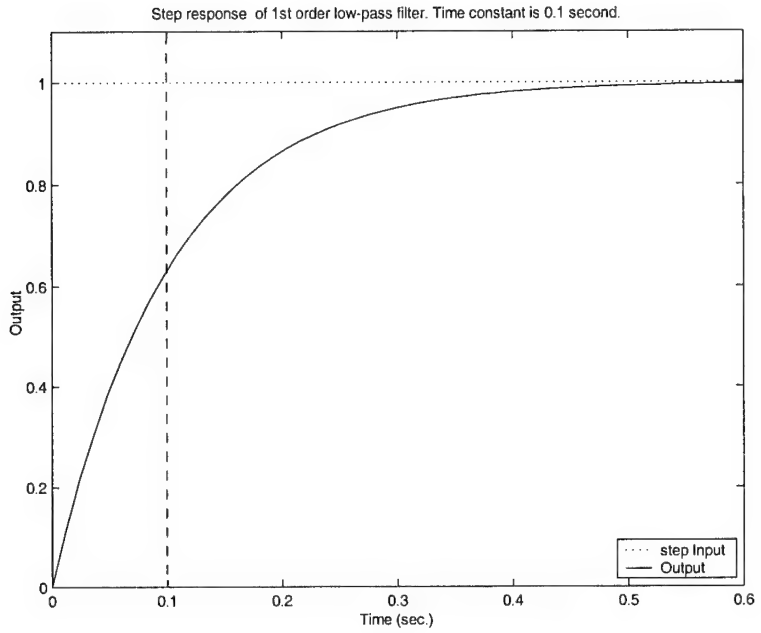


Figure 2. Step response of 1st order low-pass filter. Time constant is 0.1 second.

observed in the simulation experiments in Figure 26. Since in the conventional adaptive control paradigm the parameter is stipulated to vary "slowly", we realize that the system identification algorithm-provided parameter estimates may be far from their true value. This then motivates us to smooth the parameter estimate, viz., to pass the parameter estimate through a fixed low-pass filter. This obviously removes the fluctuations in the parameter estimate, viz., it removes the deleterious effects of noise, and by doing so, provides a better parameter estimate to be subsequently used in the compensator synthesis.

Consider the following first-order filter modeled by the scalar continuous-time state equation:

$$\dot{x}(t) = -\frac{1}{\tau}x(t) + \frac{1}{\tau}u(t) \tag{68}$$

Since our sampling time is 0.01 seconds, the required time constant is about $\tau = 0.1$. Figure 2 shows that this low-pass filter can smooth out the edge of the input signal.

Now, the discretized low-pass filter of Eq. (68) [10] is:

$$x_{k+1} = a_d x_k + b_d u_k \tag{69}$$

where

$$a_d = e^{-\frac{\triangle T}{\tau}}, \quad \triangle T = 0.01\,\text{sec.}, \quad \tau = 0.1\,\text{sec.} \tag{70}$$

$$b_d = \int_0^{\triangle T} \frac{1}{\tau} e^{-\frac{t}{\tau}} dt \tag{71}$$

and

$$a_d + b_d = 1 \tag{72}$$

Based on Eq. (68), the first-order low-pass filter used for smoothing the parameter estimate is

$$\widehat{K}_{k_{smoothed}} = \lambda \times \widehat{K}_{k-1_{smoothed}} + (1 - \lambda) \times \widehat{K}_k \tag{73}$$

where the weight λ is a fixed number and $0 < \lambda < 1$. Unfortunately, the low-pass filter inevitably introduces a lag in the parameter estimation process. This is why we are using a low order low-pass filter for smoothing the parameter estimate. The lag introduced by the fixed low-pass parameter filter is particularly problematic in reconfigurable control, where a (flight) critical parameter is subject to possibly abrupt change.

The performance of the low-pass filter for smoothing the parameter estimate and the performance of the adaptive control system which uses this filter is discussed in Chapter 7 in the sequel.

4.2 Adaptive Parameter Filter

The simplicity afforded by a fixed low-pass filter which exclusively operates on the parameter estimate provided by the system identification algorithm is appealing. Note however that the fixed low-pass filter does not use the available predicted parameter estimation error variance, which, we recall, is reliably provided by the system identification algorithm. Using the predicted estimation error variance information which is provided by our rigorous system identification algorithm, one can selectively employ smoothing. In our specific situation in which the system identification algorithm is rigorously derived and therefore the parameter estimate is not biased and the parameter estimation error variance is reliable, and when the parameter estimation error variance is small, one is justified in directly using the plant parameter estimate in the on-line compensator synthesis formula. Indeed, when the parameter estimation error variance is small we can employ *assumed certainty equivalence*, for then we know that the parameter estimate must be close to the true parameter value (only if that computed error variance is reliable). Then, there is no need to filter the parameter estimate and therefore the lag caused by passing the parameter estimate through a low pass filter is now removed. If however the parameter estimation error variance is large and we are not confident using the system identification - provided parameter estimate in the compensator synthesis, it is then advantageous to rely on filtering; this is tantamount to postulating that the parameter does not change much during the short time interval under consideration and therefore in the on-line compensator synthesis we partially rely on the old parameter estimate. In this case one introduces some lag.

Hence, we now make the filter dynamics dependent on the parameter estimation error variance provided by the upstream system identification module, and in doing so we adaptively filter the loop gain estimate. These insights into the estimation situation at hand suggest the following strategy:

Set the weight of the current parameter estimate provided by the system identification algorithm, $1 - \lambda_k$, to satisfy

$$\log_{10}(1 - \lambda_k) = \frac{1}{10^{(SNR_k - SNR_{k-w_l})}} \tag{74}$$

where w_l is the moving window length used in the system identification algorithm, and the Signal to Noise Ratio (SNR) at time k is defined as

$$SNR_k = 20 \log_{10}(\frac{K}{\sigma_{K_k}}) \tag{75}$$

Thus, A decrease in the SNR as moves from estimation window $k - wl$ to estimation window k has the effect of decreasing the reliance on the most recent loop gain estimate, $\widehat{K}_k$.

Hence, the adaptive filter for the parameter estimate is

$$\widehat{K}_{k_{smoothed}} = \lambda_k \times \widehat{K}_{k-1_{smoothed}} + (1 - \lambda_k) \times \widehat{K}_k \tag{76}$$

where the weight λ_k, $0 < \lambda_k < 1$, is adjusted according to

$$\lambda_k = 1 - 10^{-(\frac{\sigma_{K_k}}{\sigma_{K_{k-w_l}}})^{10}} \tag{77}$$

and where σ_{K_k} is the predicted parameter (K) estimation error variance provided by the system identification module at time instant k.

The adaptive parameter estimate filter, Eqs. (76) and (77), automatically smoothes the parameter estimate and removes bursting. The performance of the adaptive filter for smoothing the parameter estimate and the performance of the adaptive and reconfigurable control system which uses this filter is discussed in Chapter 7 in the sequel.

4.3 Discussion

In the conventional adaptive control paradigm one assumes that while the plant parameter is not known, is time dependent, and is subject to change and therefore needs to be identified on-line, the parameter changes slowly relative to the plant dynamics. Reconfigurable control takes adaptive control to a higher level, and allows for abrupt changes in the parameter, as would be the case under plant failure conditions. Hence, the reduction of the lag in the parameter estimate is vary relevant to reconfigurable control. Moreover, reducing the lag in the parameter estimate is particularly important when the plant under consideration is open loop unstable, feedback control is used for stabilization, and the parameter under consideration is the critical open loop gain. Now, the adaptive parameter estimate smoother developed herein uses all the available information on the plant parameter provided by the on-line system identification module and hence the lag and the error in the plant parameter estimate calculated by the smoother and sent to the compensator is minimized.

Hence, in this dissertation a novel adaptive and reconfigurable control architecture is developed which implements an automatic anti-bursting measure that entails an adaptive plant parameter smoother in tandem with the system identification module. In summary, a three-module adaptive and reconfigurable digital controller consisting of a system identification algorithm, an adaptive parameter estimate low-pass filter, and an on-line PI compensator synthesis formula, is developed.

V. *Tracking Controller*

In flight control, the aircraft physical system can be adequately represented as a mathematical model and one can synthesize a model-based feedback controller. The control law used entails full state feedback [10]. Full state feedback is indeed possible because in the controller design we use a low order truth model of the plant.

Although our indirect adaptive control approach uses a stochastic dynamic model to account properly for measurement noise, it is nevertheless desirable to consider a deterministic model first for the purpose of control design. Also, the clean states are calculated deterministically, for later comparison with the algorithm's provided estimates when noise is included in the simulation.

In this dissertation, special attention is given to the design of the (linear) tracking controller so that carefully thought out adaptive and reconfigurable flight control experiments can be performed. In this chapter, we introduce a new design methodology to design the tracking controller and analyze the effect of control surface failure on the tracking controller. Our simulation experiments validate the benefits of adaptive control, above and beyond the benefits of conventional feedback control.

5.1 Tracking Controller Design in State Space

Consider the following continuous-time m-input, l-output plant

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx = \begin{pmatrix} D \\ E \end{pmatrix} x \end{aligned}$$

where $x \in \Re^n$, $A \in \Re^{n \times n}$, $B \in \Re^{n \times m}$, $u \in \Re^m$, $y \in \Re^l$, $C \in \Re^{l \times n}$, $D \in \Re^{m \times n}$, $E \in \Re^{p \times n}$, and $m + p = l$. Let $w = Dx$ be an $m \times 1$ vector representing the outputs that are required to follow an $m \times 1$ reference signal r [9]. In steady state, we would

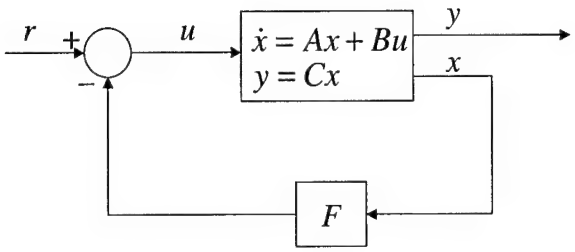


Figure 3. Conventional Tracking Control System

like to have

$$w = r, \quad \forall r \in R^m.$$

5.1.1 Conventional Tracking Controller. The conventional tracking control law is

$$u = r - Fx, \quad F \in \Re^{m \times n} \tag{78}$$

where F is a stabilizing feedback matrix. A conventional closed loop tracking control system is shown in Figure 3. Thus, the closed-loop system is

$$\begin{aligned} \dot{x} &= (A - BF)x + Br \\ y &= Cx. \end{aligned}$$

Indeed, there is an inordinate amount of attention given to stabilizing feedback control laws of the form (78) in the control literature.

The steady state analysis of the tracking control law (78) is now performed:

$$\begin{aligned} \overline{x} &= -(A - BF)^{-1}Br \\ \overline{y} &= C\overline{x} \\ &= -\begin{pmatrix} D \\ E \end{pmatrix}(A - BF)^{-1}Br \end{aligned} \tag{79}$$

where the barred quantities represent steady state values. We want asymptotic tracking,

$$\overline{w} = D\overline{x} = r \tag{80}$$

i.e.,

$$r = -D(A - BF)^{-1}Br \ , \quad \forall r \tag{81}$$

which yields the requirement

$$D(A - BF)^{-1}B = -I_m. \tag{82}$$

Hence, not only must the matrix F be a stabilizing feedback, but it must also be chosen such that the tracking condition (82) holds, which is quite a restriction.

5.1.2 Alternative Tracking Controller. Use the tracking proportional controller

$$u = K_r r - Fx$$

where $K_r \in \Re^{m \times m}$ is the reference input signal gain.

Now, the steady state analysis yields the tracking condition

$$K_r = -(D(A - BF)^{-1}B)^{-1}. \tag{83}$$

We now have the freedom to chose the feedback matrix F by assigning the poles of the closed-loop system matrix arbitrarily, provided, of course, that the poles locate in the left-half complex plane; in other words, we can chose the natural frequency and thus, the bandwidth of the (closed-loop) tracking loop.

5.1.3 PI Tracking Controller. A PI controller can work by increasing the system type without significantly changing the dominant roots of the characteris-

tic equation. This is done by placing a zero close to the origin to counteract the integrator pole located at the origin.

In order to include integral action in state space, we need to augment the plant:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ \dot{z} &= r - Dx \\ y &= Cx = \begin{pmatrix} D \\ E \end{pmatrix} x \end{aligned}$$

where $z \in \Re^m$ is the "charge" on the integrator.

Use the linear tracking control law

$$u = r - K_x x - K_z z$$

where

$$F = \begin{pmatrix} K_x, & K_z \end{pmatrix}, \quad K_x \in \Re^{m \times n}, \quad K_z \in \Re^{m \times m}.$$

Hence, the augmented closed-loop dynamics are

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{z} \end{pmatrix} &= \begin{pmatrix} A - BK_x & -BK_z \\ -D & 0 \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix} + \begin{pmatrix} B \\ I_m \end{pmatrix} r \\ y &= \begin{pmatrix} C & 0 \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix} = \begin{pmatrix} D & 0 \\ E & 0 \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix}. \end{aligned} \tag{84}$$

According to Eq. (82), we need

$$\begin{pmatrix} D & 0 \end{pmatrix} \begin{pmatrix} A - BK_x & -BK_z \\ -D & 0 \end{pmatrix}^{-1} \begin{pmatrix} B \\ I_m \end{pmatrix} = -I_m. \tag{85}$$

First, we need to chose the proportional feedback gain K_x to render $A - BK_x$ stable. In addition, we want to chose the integral gain K_z such that the augmented system matrix is stable, and thus invertible, and, in addition, Eq. (85) holds.

In fact, the following holds:

$$A - BK_x \text{ is stable } \implies A - BK_x \text{ is invertible}$$

We require the following

Lemma 5 *Consider the partitioned matrix* $\begin{pmatrix} M & N \\ P & 0 \end{pmatrix}$ *where* N *is a* $n \times m$ *matrix,* P *is a* $l \times n$ *matrix, and* M *is a* $n \times n$ *square and invertible matrix. The inverse of the partitioned matrix is*

$$\begin{pmatrix} M & N \\ P & 0 \end{pmatrix}^{-1} = \begin{pmatrix} M^{-1} - M^{-1}N(PM^{-1}N)^{-1}PM^{-1} & M^{-1}N(PM^{-1}N)^{-1} \\ (PM^{-1}N)^{-1}PM^{-1} & -(PM^{-1}N)^{-1} \end{pmatrix}.$$

□

The following holds.

Theorem 6 *Asymptotic tracking is guaranteed with integral action, provided that we chose a stabilizing proportional feedback* K_x *and a non-zero integral gain* K_z *s.t. the closed-loop system matrix is stable.*

Proof:

We apply Lemma 5 with

$$\begin{aligned} M &= A_{cl} = A - BK_x \\ N &= -BK_z \\ P &= -D \end{aligned}$$

and we use Eq. (85). Hence, the inverse of the augmented matrix that features in the "tracking condition", Eq. (85), is

$$\begin{aligned}
&\begin{pmatrix} A_{cl} & -BK_z \\ -D & 0 \end{pmatrix}^{-1} \\
= &\begin{pmatrix} A_{cl}^{-1} - A_{cl}^{-1}BK_z(DA_{cl}^{-1}BK_z)^{-1}DA_{cl}^{-1} & -A_{cl}^{-1}BK_z(DA_{cl}^{-1}BK_z)^{-1} \\ -(DA_{cl}^{-1}BK_z)^{-1}DA_{cl}^{-1} & -(DA_{cl}^{-1}BK_z)^{-1} \end{pmatrix}.
\end{aligned}$$

Applying the tracking condition, Eq. (85), we calculate

$$\begin{aligned}
&\begin{pmatrix} D & 0 \end{pmatrix}\begin{pmatrix} A_{cl} & -BK_z \\ -D & 0 \end{pmatrix}^{-1}\begin{pmatrix} B \\ I_m \end{pmatrix} \\
= &\begin{pmatrix} DA_{cl}^{-1} - DA_{cl}^{-1}BK_z(DA_{cl}^{-1}BK_z)^{-1}DA_{cl}^{-1} \\ -DA_{cl}^{-1}BK_z(DA_{cl}^{-1}BK_z)^{-1} \end{pmatrix}^T \times \begin{pmatrix} B \\ I_m \end{pmatrix} \\
= &\begin{pmatrix} 0 & -I_m \end{pmatrix}\begin{pmatrix} B \\ I_m \end{pmatrix} \\
= &-I_m
\end{aligned} \tag{86}$$

as required. Eq. (86) show that the "tracking condition" for PI control always holds, no matter what the feedback control gains K_x and K_z are, provided that K_x and K_z stabilize the augmented dynamics matrix.

□

Remark 7 *The non-zero integral gain K_z obviously influences the closed-loop dynamics of the tracking control system; in other words, how fast we approach the asymptote. Moreover, the closed-loop system needs to be stable for the tracking condition to apply.*

□

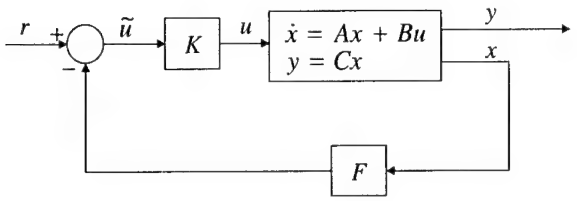


Figure 4. Tracking Control System with Variable Open-Loop Gain K

5.1.4 Tracking under Failure. We now consider the continuous-time m-input, l-output plant with variable open-loop gain matrix K - see, e.g., Fig. 4:

$$\begin{aligned} \dot{x} &= Ax + Bu = Ax + BK\widetilde{u} \\ y &= Cx = \begin{pmatrix} D \\ E \end{pmatrix} x \end{aligned}$$

where

$$K = \begin{pmatrix} k_1 & 0 & \cdots & 0 \\ 0 & k_2 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & k_m \end{pmatrix}$$

is an $m \times m$ diagonal matrix, and k_i is the degree of failure for different control surfaces.

For an unfailed control system, the open-loop gain matrix K remains an identity matrix, i.e., k_1, k_2,..., $k_m = 1$. If the control surfaces i fails at time t_f, the control derivative is reduced, and $0 < k_i < 1$, i.e., the open-loop gain matrix K is no longer an identity matrix. When B is changed to BK, Eq. (79), and consequently Eq. (83) become

$$\overline{y} = -\begin{pmatrix} D \\ E \end{pmatrix} (A - BKF)^{-1} BKr$$

and

$$K_r = -(D(A - BKF)^{-1} BK)^{-1}$$

respectively. When failure occurs, the open-loop gain changes, viz., $0 < k_1$, k_2,..., $k_m < 1$.. Then, with a set state feedback gain F we get from the "tracking condition", Eq. (82), and using Eqs. (80), (81) and (83),

$$\begin{aligned} \overline{w} &= -D(A - BKF)^{-1}BKr \\ &\neq -D(A - BF)^{-1}Br = r \end{aligned}$$

and

$$\begin{aligned} K_r &= -(D(A - BKF)^{-1}BK)^{-1} \\ &\neq -(D(A - BF)^{-1}B)^{-1}. \end{aligned}$$

Thus, the conventional proportional tracking controller can no longer track the reference (command) signal correctly.

Now, include a variable K in our PI tracking control system. Then Eq. (84) becomes

$$\begin{pmatrix} \dot{x} \\ \dot{z} \end{pmatrix} = \begin{pmatrix} A - BKK_x & -BKK_z \\ -D & 0 \end{pmatrix} \begin{pmatrix} x \\ z \end{pmatrix} + \begin{pmatrix} BK \\ I \end{pmatrix} r$$

and the tracking condition, Eq. (85), becomes

$$\begin{aligned} & \begin{pmatrix} D & 0 \end{pmatrix} \begin{pmatrix} A_{clK} & -B_K \\ -D & 0 \end{pmatrix}^{-1} \begin{pmatrix} BK \\ I \end{pmatrix} \\ = & \begin{pmatrix} DA_{clK}^{-1} - DA_{clK}^{-1}B_K(DA_{clK}^{-1}B_K)^{-1}DA_{clK}^{-1} \\ -DA_{clK}^{-1}B_K(DA_{clK}^{-1}B_K)^{-1} \end{pmatrix}^T \times \begin{pmatrix} BK \\ I \end{pmatrix} \\ = & \begin{pmatrix} 0 & -I \end{pmatrix} \begin{pmatrix} BK \\ I \end{pmatrix} \\ = & -I \end{aligned} \tag{87}$$

where

$$\begin{aligned} A_{clK} &\equiv A - BKK_x \\ B_K &\equiv BKK_z. \end{aligned}$$

Thus, Eq. (87) shows that, even when the open loop gain matrix is allowed to vary, the "tracking condition" will always hold when PI control is employed, irrespective of the open loop plant gain K and the feedback control gains K_x and K_z, provided that stability is preserved. Asymptotic tracking is guaranteed; however, in the face of control surface failure, the tracking performance suffers.

In summary, whereas the conventional tracking controller and the proportional controller can no longer maintain asymptotic tracking performance in the face of control surface failure, a PI controller will. When control surface failure occurs, the fixed PI controller can preserve asymptotic tracking performance. The proportional and integral gains need to be chosen such that closed-loop stability is universally achieved, so that the asymptotic tracking result applies. Evidently, nothing is said here about transient performance.

VI. Aircraft Model and Fixed PI Controller

In this research, an F-16 class aircraft flying at Mach 0.9 at 20,000 feet is considered. The short period pitch dynamics approximation is used. The pitch dynamics are unstable and hence the aircraft relies on feedback control for stabilization. The relevant states are α and q, the aircraft angle of attack and pitch rate, respectively, and the control variable is the elevator deflection δ_e. Thus, the plant truth model used in the system identification algorithm is

$$\dot{\alpha} = Z_\alpha \alpha + Z_q q + K Z_{\delta_e} \delta_e \tag{88}$$

$$\dot{q} = M_\alpha \alpha + M_q q + K M_{\delta_e} \delta_e. \tag{89}$$

The Z stability and control derivatives are

$$Z_\alpha = -1.3433, \quad Z_q = 0.9946, \quad Z_{\delta_e} = -0.1525$$

and the M stability and control derivatives are

$$M_\alpha = 3.5, \quad M_q = -1.0521, \quad M_{\delta_e} = -24.3282.$$

Hence, in (continuous-time) state space form, the bare aircraft (plant) dynamics are

$$\begin{aligned} \dot{x} &= Ax + bu \\ &= \begin{pmatrix} -1.3433 & 0.9946 \\ 3.5 & -1.0521 \end{pmatrix} x + \begin{pmatrix} -0.1525 \\ -24.3282 \end{pmatrix} u \end{aligned} \tag{90}$$

where the state

$$x = \begin{pmatrix} \alpha \\ q \end{pmatrix}$$

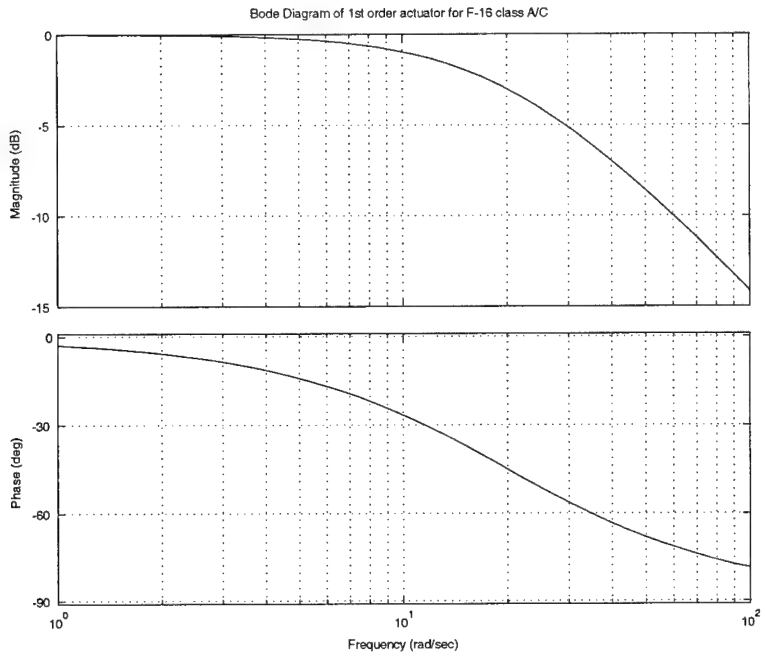


Figure 5. Bode Diagram of firstst-order actuator for F-16 class A/C.

and the control variable u is the elevator deflection δ_e. The above second-order plant model is the truth model used in the system identification algorithm.

In the linear tracking controller, the reference signal r is summed with the states α and q feedback. The controller-generated command to the elevator, δ_{e_c}, is applied to a first order actuator model $\frac{20}{S+20}$ with a bandwidth of 20 rad/sec. The actuator output, δ_e, is the input to the plant. A first-order actuator model suffices in the "low frequency" bandwidth of the pitch dynamics as shown in Figure 5. A more elaborate fourth-order actuator model [21] is

$$\frac{\delta_e(S)}{\delta_{ec}(S)} = \frac{(20.2)(5097.96)(144.8)}{(S+20.2)(S^2+1008S+5097.96)(S+144.8)} \tag{91}$$

and its Bode plot is shown in Figure 6. However the first-order actuator model captures the lag characteristics of the actuator in the bandwidth of interest.

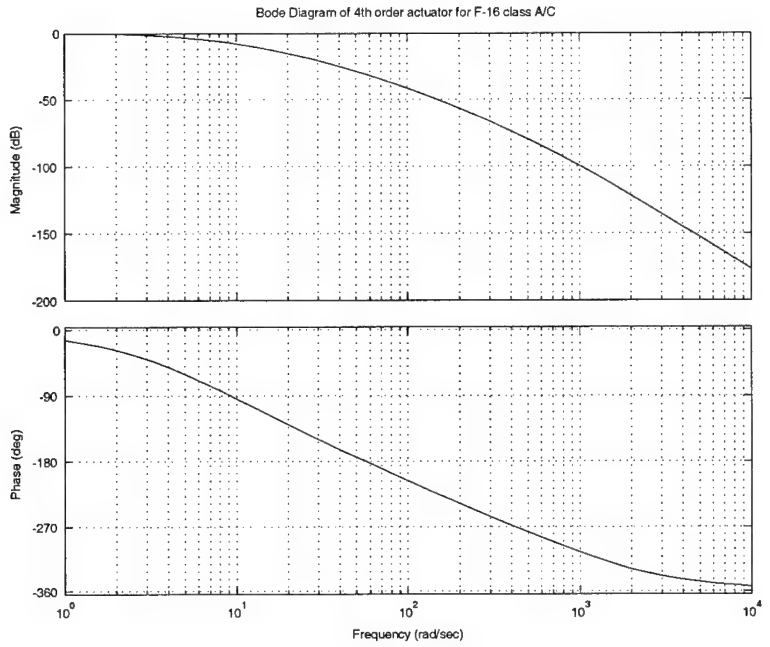


Figure 6. Bode Diagram of fourth order actuator for F-16 class A/C.

Augmenting the dynamics and control matrices with the first-order actuator dynamics yields the third-order augmented plant

$$\begin{aligned} \dot{x} &= Ax + bu \\ &= \begin{pmatrix} Z_\alpha & Z_q & Z_{\delta_e} \\ M_\alpha & M_q & M_{\delta_e} \\ 0 & 0 & -\frac{1}{\tau} \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix} \delta_{e_c}. \end{aligned} \tag{92}$$

Now the states are

$$x = \begin{pmatrix} \alpha \\ q \\ \delta_e \end{pmatrix}. \tag{93}$$

The control signal is conventionally generated according to

$$\delta_{e_c} = r - (K_\alpha \quad K_q \quad K_{\delta_e})x \tag{94}$$

Table 1. Longitudinal Dynamics with simple Tracking Controller.

System	Eigenvalues		
Open-loop	-20.0	-3.0691	0.6737
Closed-loop, K=1	-14.4615+15.5040j	-14.4615-15.5040j	-1.4724
Closed-loop, K=0.8	-1434894+12.4774j	-1434894-12.4774j	-1.4165
Closed-loop, K=0.4	-18.0087	-11.2516	-1.1351
Closed-loop, K=0.2	-24.3672	-5.3706	-0.6576
Closed-loop, K=0.09	-26.5020	-3.8255	-0.0679
Closed-loop, K=0.08	-26.6776	-3.7210	0.0032

and the first order actuator dynamics is

$$\dot{\delta}_e = -\frac{1}{\tau}\delta_e + \frac{1}{\tau}\delta_{e_c} \tag{95}$$

where $\tau = 0.05$ sec. The above third-order plant model is the truth model used for tracking controller design. The longitudinal dynamics are given in Table 1.

The closed loop dynamics A matrix is now formed using the state feedback tracking control law in Eq. (94). The resulting A and b closed loop matrices are

$$\begin{aligned}
A_{cl} &= A + b\begin{pmatrix} -K_\alpha & -K_q & -K_{\delta_e} \end{pmatrix} \\
&= \left(\begin{pmatrix} Z_\alpha & Z_q & Z_{\delta_e} \\ M_\alpha & M_q & M_{\delta_e} \\ 0 & 0 & -\frac{1}{\tau} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix} \begin{pmatrix} -K_\alpha & -K_q & -K_{\delta_e} \end{pmatrix} \right) \\
&= \begin{pmatrix} Z_\alpha & Z_q & Z_{\delta_e} \\ M_\alpha & M_q & M_{\delta_e} \\ -\frac{1}{\tau}K_\alpha & -\frac{1}{\tau}K_q & -\frac{1}{\tau}(1+K_{\delta_e}) \end{pmatrix} \\
B_{cl} &= \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix}
\end{aligned} \tag{96}$$

so we get

$$\dot{x} = A_{cl}x + B_{cl}r. \tag{97}$$

This proportional controller does not in general yield asymptotic tracking. Hence, it is modified in the sequel.

The F-16 class plant is open-loop unstable. This is a normal characteristic of advanced fighter aircraft. Using full state feedback, the flight control system is stabilized. The ensuing closed loop linear state feedback control system is very robust; a well designed simple tracking controller, Eq. (94), can handle open-loop gains as low as $K_1 = 0.08$ while preserving stability, although tracking performance is significantly degraded after the degree of failure increases to $K_1 = 0.2$. At lower K_1 values, the feedback stabilization action becomes ineffective and the closed loop system becomes unstable, as shown in Table 1 and Figure 7. In this work, two tracking controllers are considered, a fixed proportional controller and a fixed Proportional plus Integral (PI) controller.

6.1 Proportional Controller

The tracking controller is now designed. Based on it, a proportional controller and an alternative fixed PI controller will be designed in the following sections.

To find the gain needed to improve tracking performance, the augmented closed-loop state space equation is used. The reference signal $r(\equiv q)$ is the exogenous input and control signal is

$$\delta_{e_c} = K_r r - (K_\alpha \alpha + K_q q + K_{\delta_e}\delta_e). \tag{98}$$

At steady state, $\dot{x} = A_{cl}x + K_r B_{cl}r = 0$. Writing the augmented state space equation with the necessary gain, K_r, at steady state,

$$0 = A_{cl}\overline{x} + K_r B_{cl}r \tag{99}$$

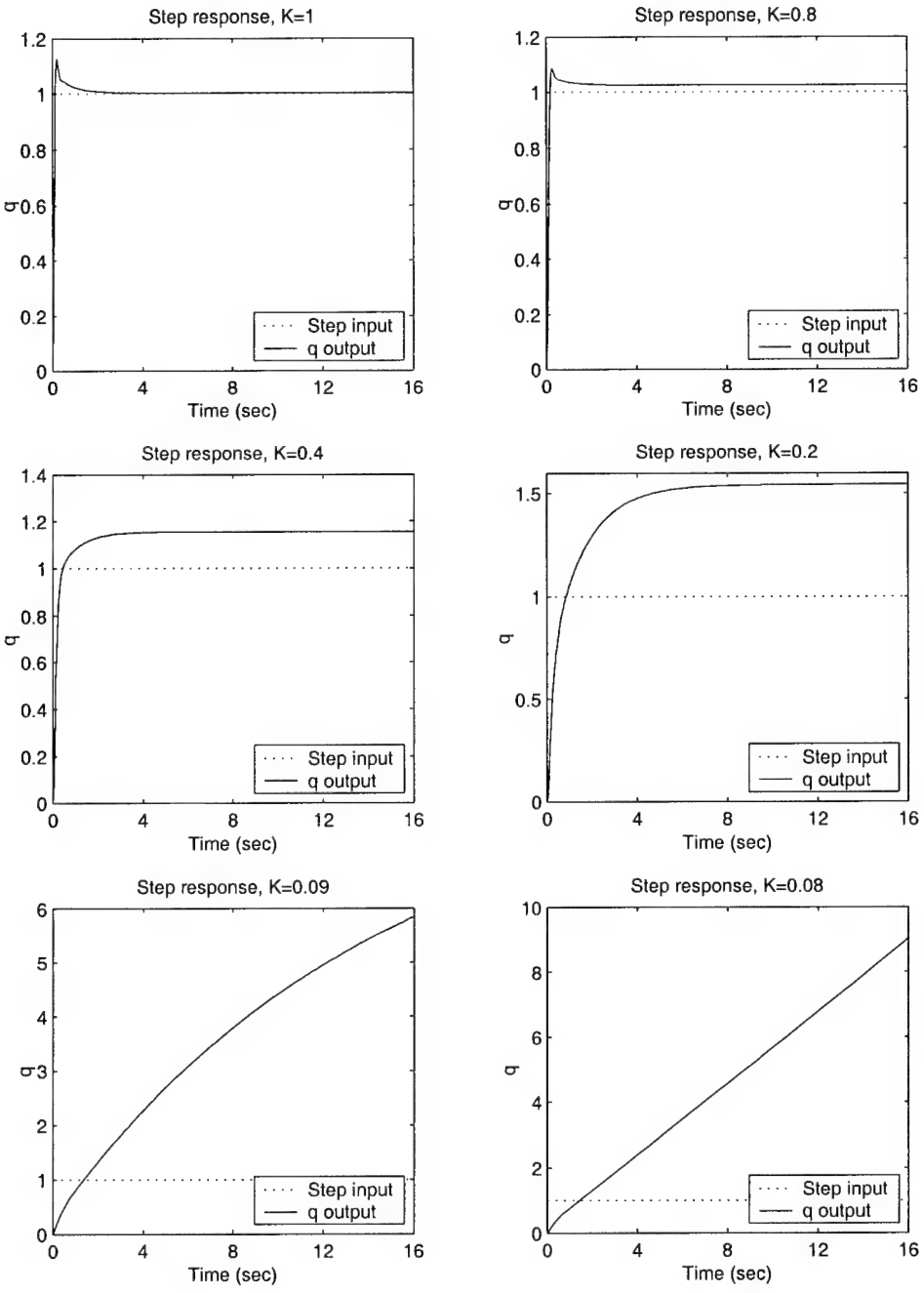


Figure 7. Simple tracking step responses with various open-loop gains K_1.

and solving for $\overline{x}$ yields

$$\overline{x} = K_r A_{cl}^{-1} B_{cl} r. \tag{100}$$

Substituting the above into the output equation

$$\overline{y} = c\overline{x} \tag{101}$$

gives us

$$\overline{y} = K_r c(A_{cl}^{-1} B_{cl} r). \tag{102}$$

It is desired for the output to track a step input. For this to happen

$$\overline{y} = r \tag{103}$$

which implies

$$1 = K_r c A_{cl}^{-1} B_{cl} \tag{104}$$

must hold. Solving for u gain K_r, yields

$$K_r = \frac{1}{c A_{cl}^{-1} B_{cl}} \tag{105}$$

which gives the required reference signal gain needed for proper tracking. This gain is then applied to the system before the feedback loop as shown in Figure 1. This adjusts our B_{cl} to the following

$$B_{cl} = \begin{pmatrix} 0 \\ 0 \\ \frac{K_r}{\tau} \end{pmatrix}. \tag{106}$$

Table 2. Longitudinal Dynamics with Proportional Tracking Controller.

System	Eigenvalues		
Open-loop	-20.0	-3.0691	0.6737
Closed-loop, K=1	-14.4615+15.504j	-14.4615-15.504j	-1.4724
Closed-loop, K=0.8	-14.4894+12.4774	-14.4894-12.4774	-1.4165
Closed-loop, K=0.4	-18.0087	-11.2516	-1.1351
Closed-loop, K=0.2	-24.3672	-5.3706	-0.6576
Closed-loop, K=0.09	-26.5020	-3.8255	-0.0679
Closed-loop, K=0.08	-26.6776	-3.7211	0.0032

The gains needed for K_α, K_q and K_{δ_e} are

$$\begin{aligned} K_\alpha &= 0.283 \\ K_q &= 0.876 \\ K_{\delta_e} &= -0.4. \end{aligned}$$

So, the proper tracking dynamics are obtained.

Tracking is achieved using a fixed proportional controller. Table 2 shows the eigenvalues of the closed-loop system using the fixed proportional tracking controller.

Once again, the open-loop plant is unstable and feedback stabilization is used. The closed-loop system becomes unstable again when the degree of control surface loss becomes excessively large, viz., $K_1 = 0.08$ as shown in Table 2 and Figure 8. However, the tracking performance degrades significantly after the degree of failure increases to $K_1 = 0.2$.

6.2 PI Controller

Designing a PI controller in state space for good tracking performance requires the system dynamics to be further augmented. Now

$$\dot{z} = r - q \tag{107}$$

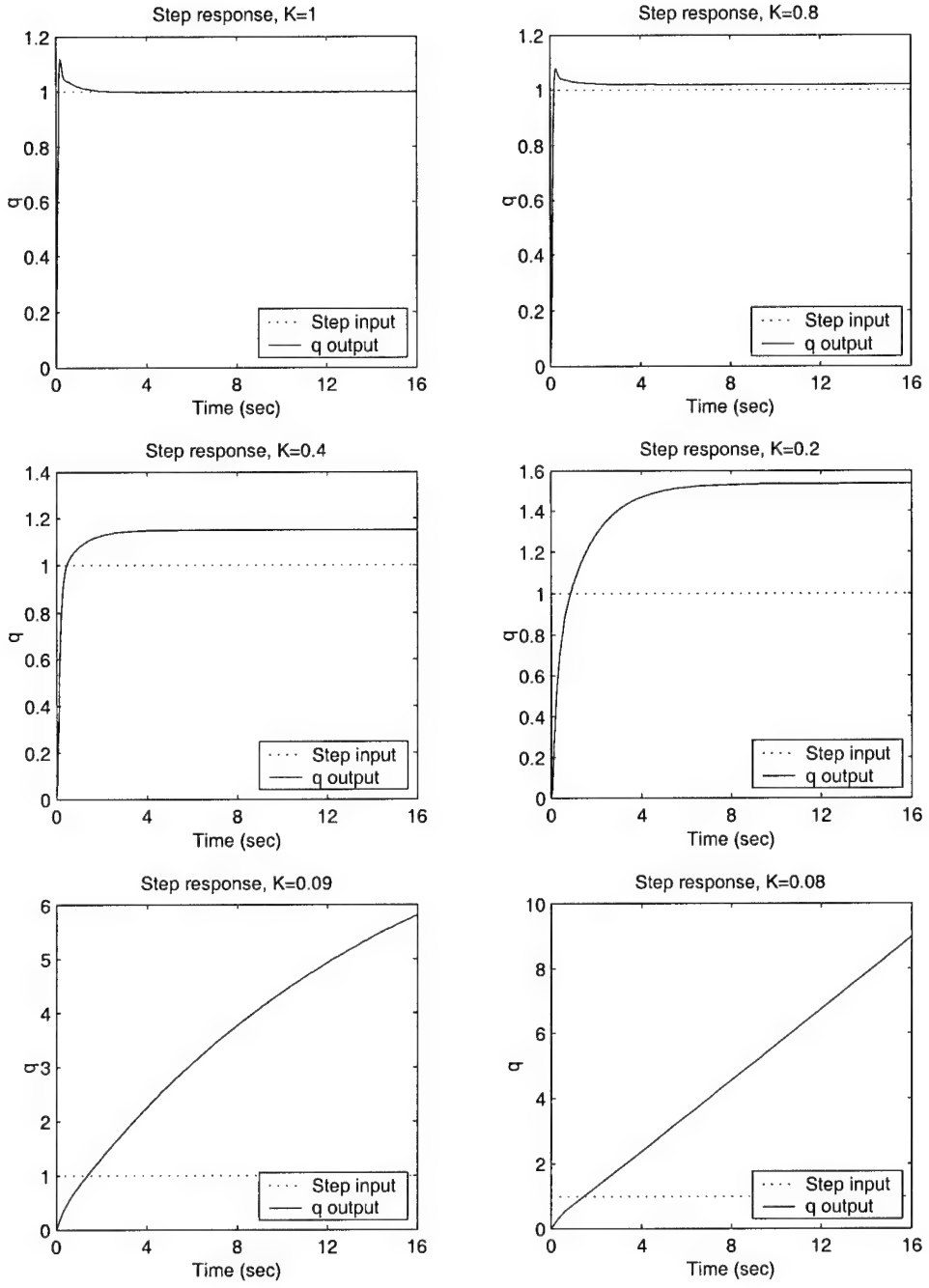


Figure 8. Proportional controller step responses with various open-loop gains K_1.

and

$$\dot{x} = \begin{pmatrix} Z_\alpha & Z_q & Z_{\delta_e} & 0 \\ M_\alpha & M_q & M_{\delta_e} & 0 \\ 0 & 0 & -\frac{1}{\tau} & 0 \\ 0 & -1 & 0 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} r + \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \\ 0 \end{pmatrix} \delta_{e_c} \tag{108}$$

where the states are now

$$x = \begin{pmatrix} \alpha \\ q \\ \delta_e \\ z \end{pmatrix} \tag{109}$$

and, as before, r is the reference signal. The "charge" on the integrator is z. Now the PI control law is

$$\delta_{e_c} = r - K_\alpha \alpha - K_q q - K_{\delta_e} \delta_e + K_z z. \tag{110}$$

The new closed-loop system matrices A_{cl} and B_{cl} are

$$\begin{aligned} A_{cl} &= A + b \begin{pmatrix} -K_\alpha & -K_q & -K_{\delta_e} & K_z \end{pmatrix} \\ &= \begin{pmatrix} Z_\alpha & Z_q & Z_{\delta_e} & 0 \\ M_\alpha & M_q & M_{\delta_e} & 0 \\ -\frac{K_\alpha}{\tau} & -\frac{K_q}{\tau} & -\frac{(1+K_{\delta_e})}{\tau} & \frac{K_z}{\tau} \\ 0 & -1 & 0 & 0 \end{pmatrix} \\ B_{cl} &= \begin{pmatrix} 0 \\ 0 \\ \frac{1}{\tau} \\ 1 \end{pmatrix}. \end{aligned} \tag{111}$$

Table 3. Longitudinal Dynamics with PI Tracking Controller.

System	Eigenvalues			
Open-loop	-20.0	-3.0691	0.6737	N/A
Closed-loop, K=1	-14.4560+15.4990j	-14.4560-15.4990j	-1.4733	-0.01
Closed-loop, K=0.8	-14.4841+12.4721j	-14.4841-12.4721j	-1.4169	-0.0103
Closed-loop, K=0.4	-18.0244	-11.2266	-1.1328	-0.0116
Closed-loop, K=0.2	-24.3692	-5.3625	-0.6480	-0.0157
Closed-loop, K=0.09	-0.0352+0.0683j	-0.0352-0.0683j	-26.5027	-3.8222
Closed-loop, K=0.08	0.0005+0.0732j	0.0005-0.0732j	-26.6782	-3.7181

The proportional and integral gains needed to obtain good tracking performance are

$$\begin{aligned} K_\alpha &= 0.283 \\ K_q &= 0.876 \\ K_{\delta_e} &= -0.4 \\ K_z &= 0.01. \end{aligned} \tag{112}$$

Thus, tracking is achieved using a fixed PI controller. Table 3 shows the poles of the open-loop plant and the poles of the closed-loop system when this fixed PI controller is used for tracking control. As can be seen in Table 3 and Figure 9, the bare plant is originally open-loop unstable. State feedback stabilizes the α, q, and δ_e states of the unimpaired closed-loop flight control system. As the loop gain K is lowered from a value of 1, which corresponds to having no failure, to a value of $K_1 \approx 0.08$, an almost complete longitudinal control surface loss, the closed-loop system reverts to instability again. However, the tracking performance degrades significantly after the degree of failure increases to $K_1 = 0.2$.

In this chapter, the deterministic aircraft model of the control system is developed. It was shown how the $[A\ ,\ b]$ plant is augmented with actuator dynamics, and the tracking control law was introduced. Because the "standard" tracking controller could not properly track a step input, a proportional and a PI controller are imple-

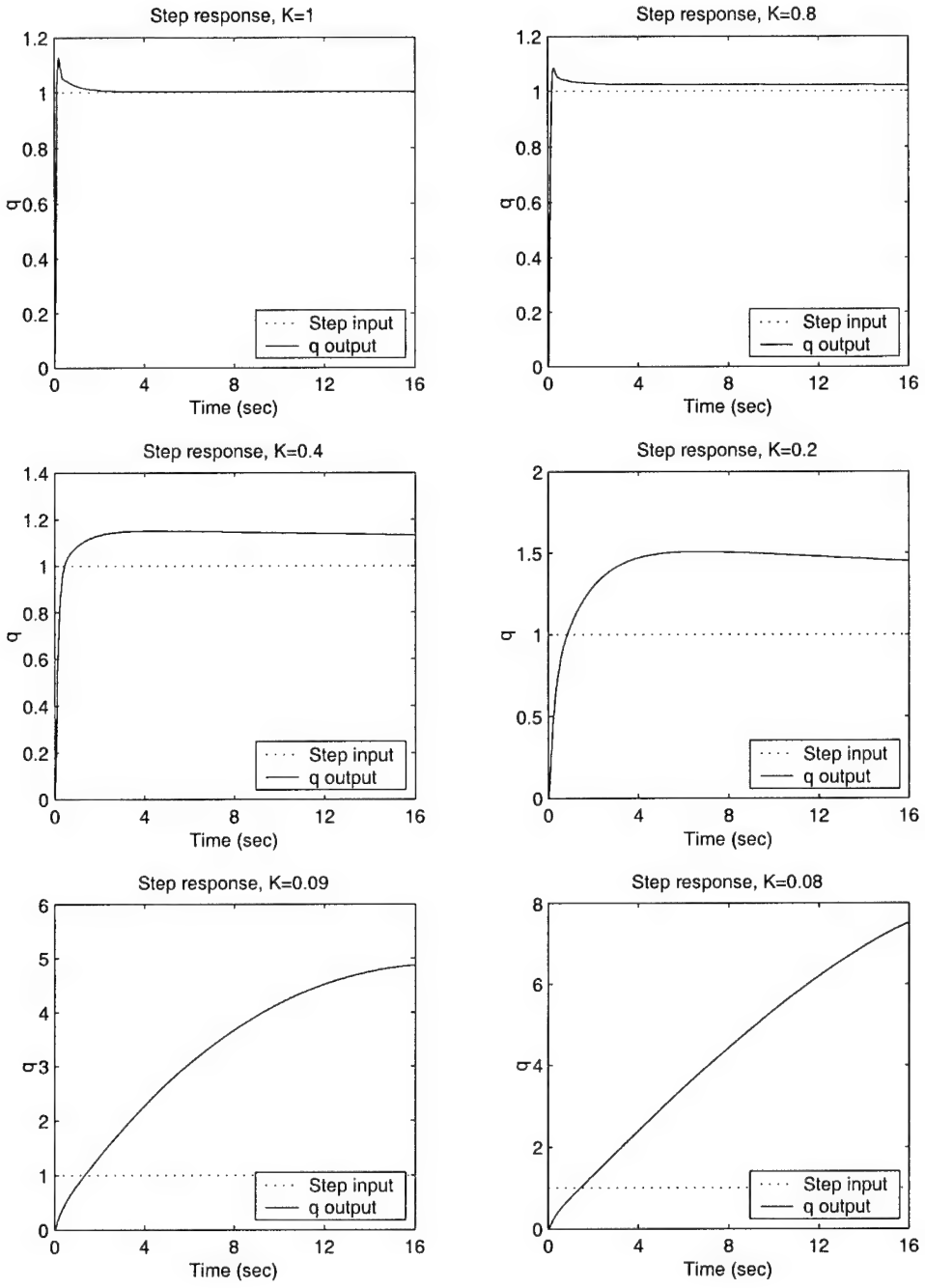


Figure 9. PI controller step responses with various open-loop gains K_1.

mented for two different case studies. The closed loop matrices were found for both fixed controllers and the gains were designed by pole assignment.

VII. Simulation Results

7.1 Experimental Setup and Tracking Control

One would like a feedback control system to be robust enough to perform within specifications in the face of parametric uncertainty, e.g., control surface loss due to failure or battle damage, and in the presence of measurement noise and unmodeled dynamics. As the critical open-loop gain K decreases from $K = 1$ to $K \ll 1$, which represents a transition from no failure to substantial control surface loss, the tracking of the reference signal slips. Even though failure, viz., a reduction in the control derivative, causes a fixed controller's tracking performance to deteriorate, still, a correct system identification algorithm will properly estimate the degree of failure. In fact, poor tracking caused by failure increases excitation, which boosts the performance of the system identification algorithm. Hence one is motivated to use the available signals required for feedback control in an on-line system identification algorithm and subsequently adjust the controller's gain on line in order to account for the failure-induced reduction in the plant's open-loop gain, thus recovering performance and achieving adaptive and reconfigurable control action. This control concept, illustrated in Figure 1, is implemented in our simulation.

In most of our MATLAB [14] simulations, the command signal consists of 4 pitch rate (q_c) doublet commands, having an amplitude of ± 10 deg/sec and a period of 4 seconds, giving a 16 seconds measurement record. The input command represents a pilot "exciting the stick" maneuver, used in flight test. The doublets are passed through a low pass prefilter, $\frac{3}{S+3}$. Such a prefilter is currently used in F-16 aircraft. The pulsed command signal and the ensuing reference pitch rate command signal are shown in Figure 10. Except when specified otherwise, a control surface failure is induced at $t_f = 8$ seconds into the flight in all the test runs. This translates into a jump in the parameter K from 1 to $K = K_1$, $0 < K_1 < 1$. In our experiments the simulated degree of failure is known and is parameterized by K_1. Thus, during

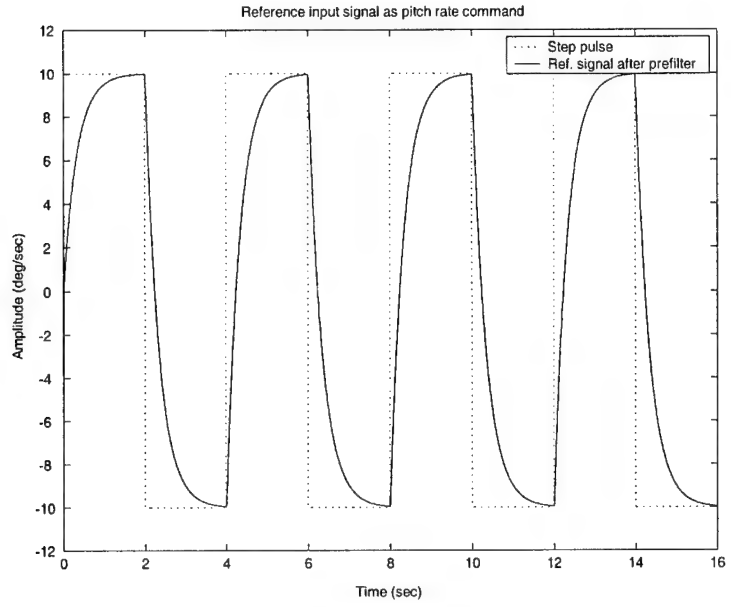


Figure 10. Reference input signal - pitch rate command. The "frequency" is $\pi/2$ rad/sec and the amplitude is 10 deg/sec.

the first 8 seconds, we simulate an unimpaired A/C, and at $t = t_f = 8$ seconds we simulate an elevator surface loss, so that for the remaining 8 seconds we control an impaired A/C. Hence, the open-loop gain

$$K(t) = \begin{cases} 1 & \text{for } 0 \leq t < 8 \\ K_1 & \text{for } 8 \leq t \leq 16 \end{cases}$$

where

$$0 < K_1 < 1.$$

In addition, measurement (sensor) noise is properly injected into the simulation. Thus, the measured pitch rate is $q_m = q + v_q$, where $v_q = N\left(0, \sigma_q^2\right)$, and the measured angle of attack is $\alpha_m = \alpha + v_\alpha$, where $v_\alpha = N(0, \sigma_\alpha^2)$. Given that both α and q observations are used for system identification, the following definition of SNR is used:

$$SNR \triangleq 20 \log \sqrt{\frac{\alpha_{\max}^2 + w^2 q_{\max}^2}{2\left(\sigma_\alpha^2 + w^2 \sigma_q^2\right)}} \tag{113}$$

where the weighting

$$w = \frac{1}{\sqrt{|p_1 p_2|}} \text{ (sec)}$$

and p_1, p_2 are the poles of the open loop plant (the short period approximation):

$$p_1 = 0.6737 \qquad p_2 = -3.0691.$$

In our simulations,

$$\alpha_{\max} \approx 7 \text{ deg.}, \qquad q_{\max} \approx 11 \; deg/sec.$$

The experimental results presented in Figures 35-24 were obtained using a fixed $\sigma_\alpha = 0.03 \; deg.$

For the SNR experiments of 40 and 60 dB, a scaled σ_q and σ_α are used. We initially let $\sigma_{q_i} = 0.05$ deg / sec, and $\sigma_{\alpha_i} = 0.05 \; deg$ and we get

$$\sigma_q = k\sigma_{q_i}, \qquad \sigma_\alpha = k\sigma_{\alpha_i} \tag{114}$$

where the SNR scaling parameter k>0. The SNR is now expressed as

$$SNR \triangleq 20 \log \sqrt{\frac{\alpha_{\max}^2 + w^2 q_{\max}^2}{2\left(\sigma_\alpha^2 + w^2 \sigma_q^2\right) k^2}} \tag{115}$$

and therefore, for a specified SNR, the parameter k is determined according to

$$k = 10^{-\frac{SNR}{20}} \times \sqrt{\frac{\alpha_{\max}^2 + w^2 q_{\max}^2}{2\left(\sigma_\alpha^2 + w^2 \sigma_q^2\right)}} \tag{116}$$

and in the simulation experiments, σ_α and σ_q are adjusted according to Eq. (114). The σ_α, σ_q and SNR values are shown in Table 4.

We inject Gaussian noise of intensity $\sigma_\alpha = 0.03$ [deg] and $\sigma_q = 0.1108$ [deg/sec] using the random numbers generator of MATLAB [14] to simulate measurement

Table 4. SNR Values.

σ_q deg / sec	σ_α deg	**SNR dB**
0.56	0.03	25.54
0.11	0.03	38.96
0.06	0.06	40
0.01	0.03	47.49
0.006	0.006	60

noise for the α and q signals. The measurement noise is propagated throughout the feedback control system. The controller's sampling rate is 100 Hz. The estimate of the open-loop gain, $\widehat{K}$, is continuously calculated by using the system identification algorithm (Theorem 1) using, in most experiments, a moving data window of length 0.3 seconds (30 samples). The estimated gain is smoothed, and, in most experiments, the gain information is used in the proposed controller at each sample time - thus achieving on-line operation of the adaptive and reconfigurable control system.

The performance of each of the three modules of the adaptive and reconfigurable controller, viz, 1) the system identification module, 2) the parameter estimate smoother, and 3) the tracking controller, is now separately assessed, followed by an evaluation of the operation of the complete adaptive and reconfigurable controller.

During the simulations of $SNR's$ effects, several SNRs listed in Table 4 are used in order to analyze the $SNR's$ effect on the estimation and tracking performance. The window size effects on the performance of our three-module adaptive and reconfigurable controller will be investigated too. For more realistic, we will also include unmodeled dynamics (Phugoid dynamics, fourth-order actuator, and parameter modeling error) into our basic model, and investigate the effects.

7.2 Estimation Performance

7.2.1 Expanding Window System Identification. The estimation performance guaranteed by the novel system identification algorithm stated in Theorem 1 is experimentally validated, and the results of the open-loop gain identification

experiments are presented. The plant truth model used in the system identification algorithm is

$$\begin{aligned} \dot{\alpha} &= Z_\alpha \alpha + Z_q q + K Z_{\delta_e} \\ \dot{q} &= M_\alpha \alpha + M_q q + K M_{\delta_e} \end{aligned}$$

viz., the state $\begin{pmatrix} \alpha \\ q \end{pmatrix}$ evolves in $\Re^2$ and the control signal is $u = \delta_e$. A discrete-time version of the plant which corresponds to a sampling rate of $100Hz$ is embedded in the system identification algorithm. In our experiments, the quality of the prior information given to the system identification algorithm is intentionally chosen to be poor. It therefore takes the system identification algorithm some time to settle down and output the correct parameter estimates.

Figures 11 - 14 show how the system identification algorithm, using an expanding horizon Kalman filter (Theorem 1), estimates the open-loop gain K as the degree of failure increases. The true open-loop gain K and the identified open-loop gain $\widehat{K}$ when using the fixed PI tracking controller mechanization are shown. The parameter estimate shown are output by the expanding window system identification algorithm and the parameter estimate smoother is not used. The settling time is fairly long. However, the settling time is shortened when K_1 is small. This is clearly a nonlinear phenomenon. Also note the large parameter estimation errors (spikes) near $t = 0$, before the estimation window fills up; also, it is evident that the initialization transient terminates at time t $\approx$ 4 sec.

The estimation performance for various degrees of failure, and the estimation time delay defined when the K estimate is within 10% of the true K after failure (K_1), is summarized in Table 5.

Table 5. Open-loop gain estimation performance with expanding data window. Fixed PI tracking controller and no parameter estimate smoother.

Actual Post-Failure Open-Loop Gain K_1	Final Est. Value $\widehat{K}$	Relative Est. Error(%)	Time Delay (sec.)
0.8	0.8878	10.98	8.52
0.6	0.7311	21.86	11.81
0.4	0.4602	15.06	12.65
0.2	0.2073	3.63	4.77
0.1	0.1006	0.64	2.20

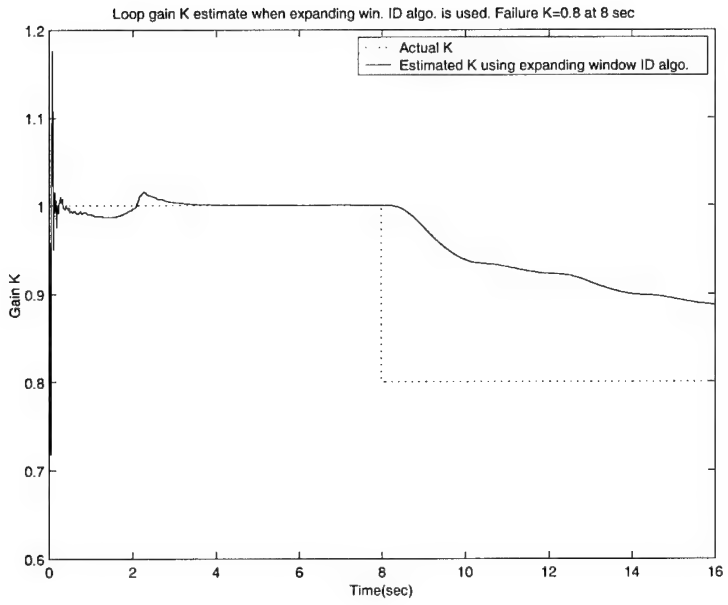


Figure 11. Open-loop gain estimation when $K_1 = 0.8$, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$ when an expanding window system ID algorithm is used.

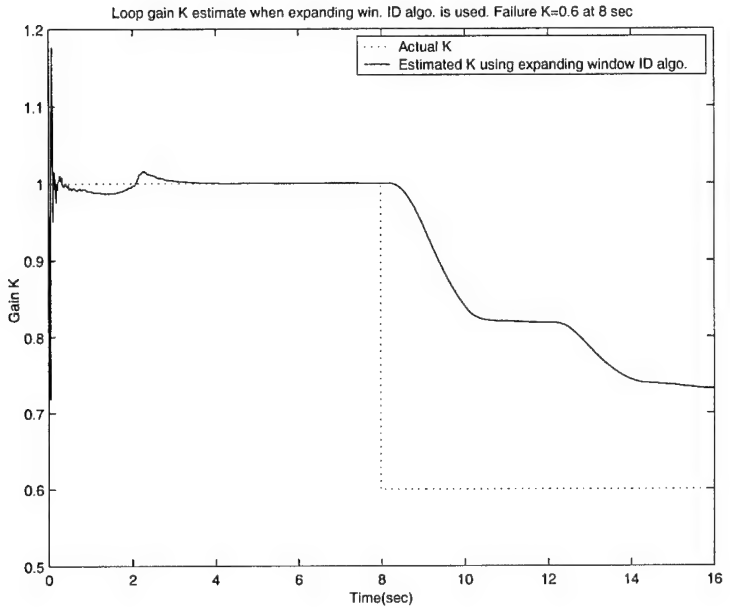


Figure 12. Open-loop gain estimation when $K_1 = 0.6$, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$ when an expanding window system ID algorithm is used.

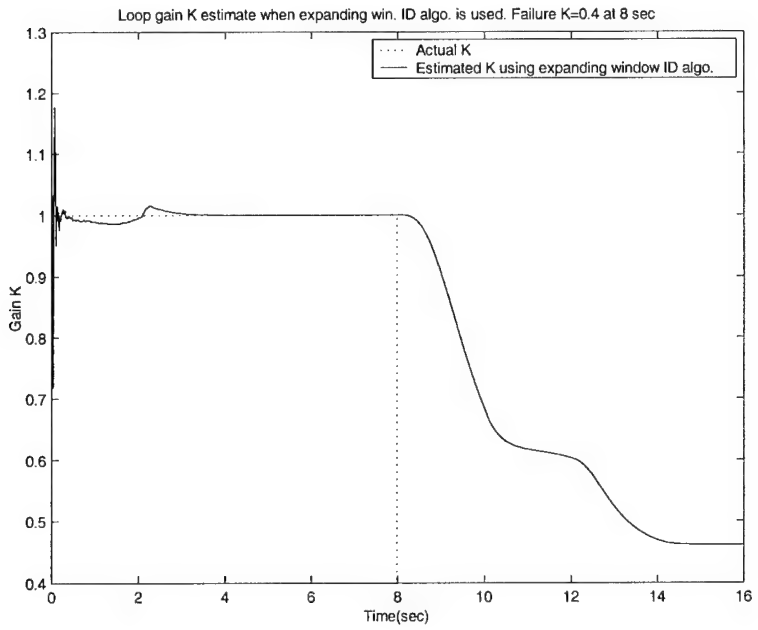


Figure 13. Open-loop gain estimation when $K_1 = 0.4$, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$ when an expanding window system ID algorithm is used.

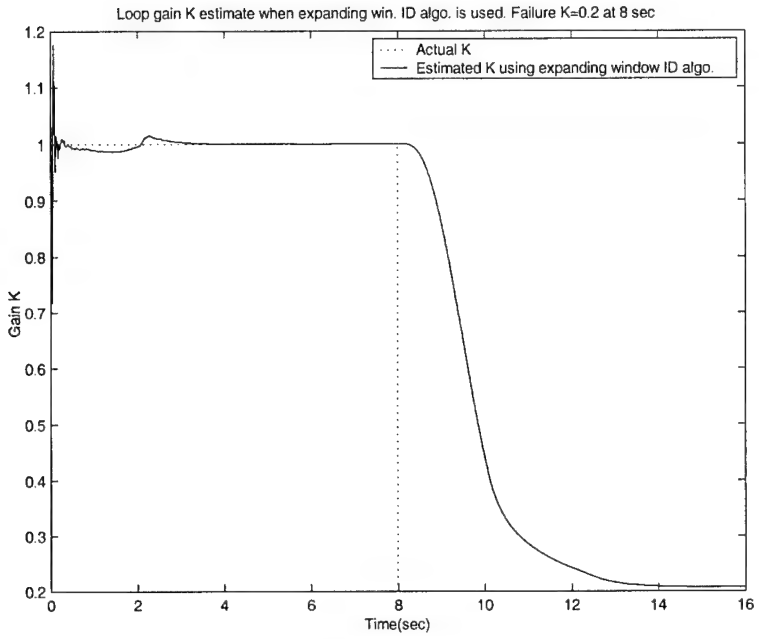


Figure 14. Loop gain estimation when $K_1 = 0.2$, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$ when an expanding window system ID algorithm is used.

7.2.2 Moving-Window System Identification. The long parameter estimation delays observed when an expanding horizon Kalman filter is used motivates us to use a moving-window filter. A moving-window (or, equivalently, finite memory data window) is preferred because, in the case of a failure, when a jump in the value of the open-loop gain K occurs, the latter is identified faster than in the case where the expanding horizon window system identification algorithm is used. By using the recursive system identification algorithm (Theorem 1 in Chapter 3) inside a 0.3 second window (of 30 samples), estimates of the parameters of interest are calculated. The window is then shifted one sample time and the estimation process is repeated. This yields the first parameter estimate at 0.3 seconds into the flight. Prior information with negative α and q states and an initial guess of $K = 0.8$ are intentionally used to test the moving-window estimation algorithm's response to a poor initial guess. For all of the windows, the same prior information of $\alpha = -1.4414$ degrees, $q = -2.4314$ degrees/second, and $K = 0.8$ is used. The initial states α and q variances are 0.1

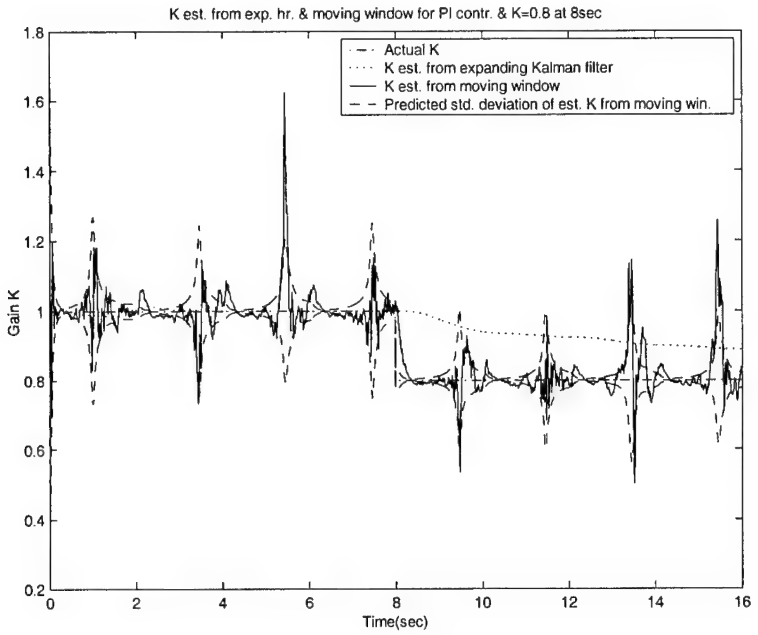


Figure 15. Comparison of expanding horizon and moving window estimation with fixed PI tracking controller. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.8$.

$[deg^2]$, 1 $[(deg/sec)^2]$, respectively, and the variance of the parameter K initial guess is 0.4.

Setting the post failure open-loop gain at $K_1 = 0.8$, $K_1 = 0.6$, $K_1 = 0.4$, and $K_1 = 0.2$, we compare the open-loop gain $(\widehat{K})$ estimation performance of the moving-window system identification algorithm and the expanding window system identification algorithm. The fixed PI tracking controller, and no parameter estimate filter, are used - see, e.g., Figures 15 to 18. One can see that the moving-window is faster to settle on an estimate, while the expanding horizon system identification algorithm takes more time to reach its final estimate value. Obviously, the estimate provided by the expanding window Kalman filter is smoother than the estimate provided by the relatively short sliding window. At the same time, the negative effect on estimation performance of a very short window ($\ll 0.3sec$) is also evident near $t = 0$.

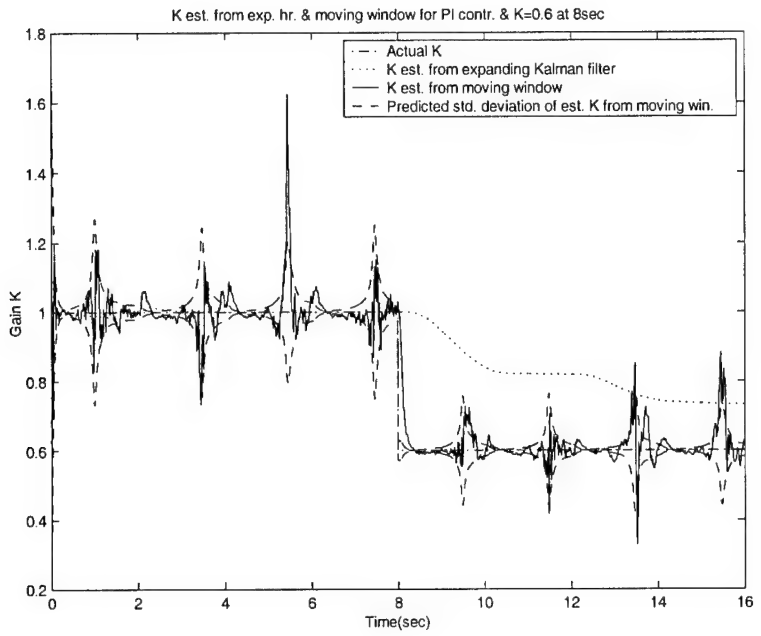


Figure 16. Comparison of expanding horizon and moving window estimation with fixed PI tracking controller. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

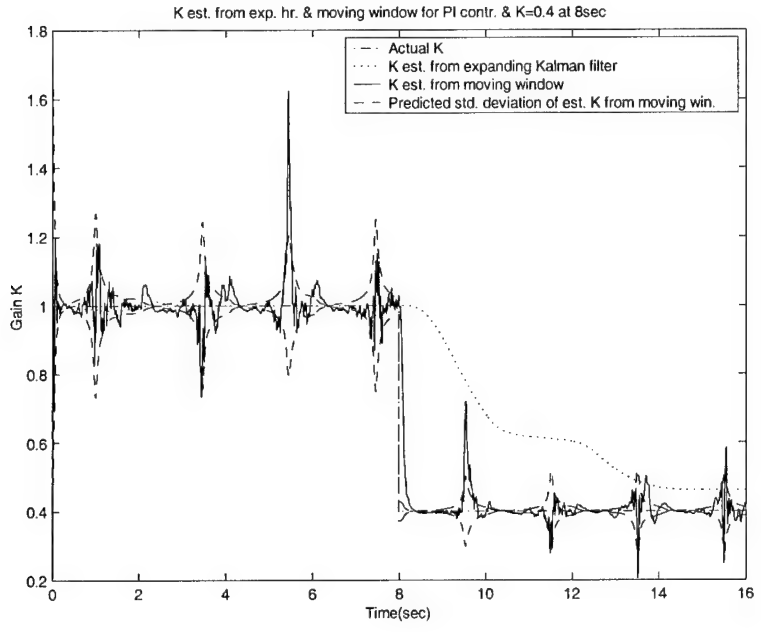


Figure 17. Comparison of expanding horizon and moving window estimation with fixed PI tracking controller. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.4$.

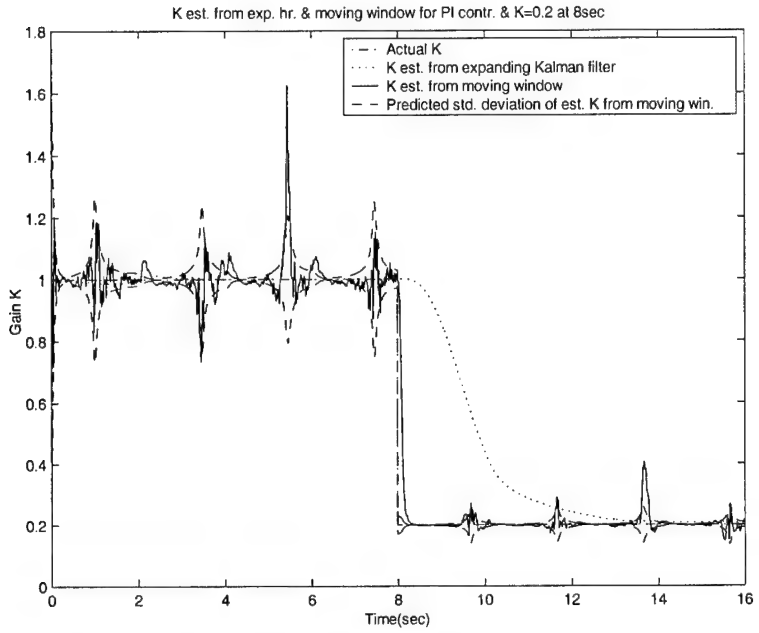


Figure 18. Comparison of expanding horizon and moving window estimation with fixed PI tracking controller. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.2$.

In Figures 15 to 18, the true parameter $\pm$ the predicted standard deviation of the parameter estimate calculated by the moving-window system identification algorithm are also shown. About 50.84%, 50.97%, 50.53%, and 51.78% of the K estimates, $\widehat{K}$, fall inside the predicted 1σ bound for $K_1 = 0.8$, $K_1 = 0.6$, $K_1 = 0.4$, and $K_1 = 0.2$, respectively. Also note that when the moving-window system identification algorithm is used, for certain data windows, the excitation in the window is poor, and these particular windows yield poor parameter estimates. At the same time, the predicted parameter estimation error's standard deviation, σ_K, is large, which shows that the system identification algorithm is performing as expected. The poor estimates occur when the window slides past the peaks of the input command, as shown in Figures 15 to 18, i.e., the spikes in the parameter estimate seem to be correlated with the input peaks, as is evident in Figure 19. The input command is then near-constant, which yields poor excitation, which is reflected in a large σ_K; this is bad for adaptive control.

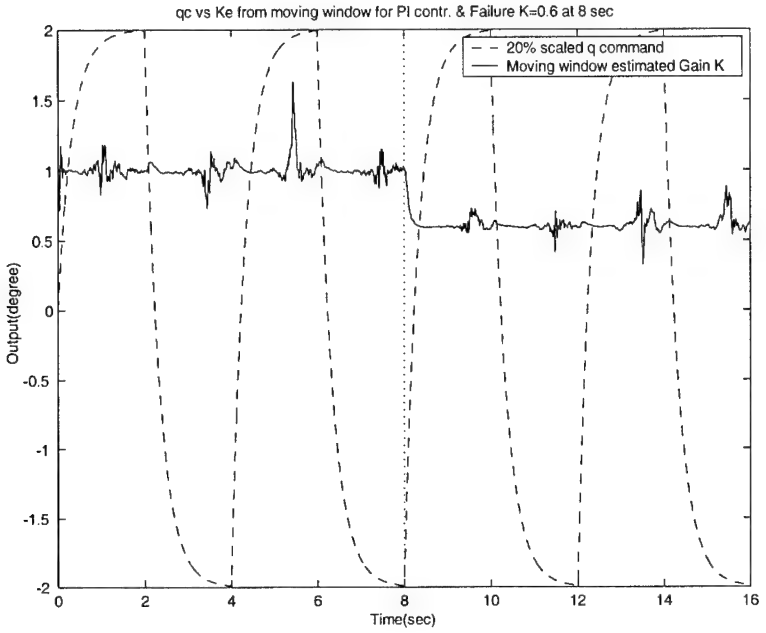


Figure 19. The reference command and the estimate $\widehat{K}$ provided by the moving window system identification algorithm. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.

In this regard, Figures 20 and 21 show the estimation performance and tracking performance when the command signal is a unit step function, i.e., a 1 deg/sec pitch rate step command. Although now, due to the low SNR, and low excitation, the parameter estimate is not so good and the estimation error variance is high, nevertheless, about 69.27% of the K estimates, $\widehat{K}$, fall inside the predicted 1σ bound of $\widehat{K}$. Evidently, the theoretical score is 68.3% - which, again shows that the system identification algorithm performs as expected.

We finally note that, good estimation performance is recorded when the amplitude of the input step is 10 deg/sec. This is due to the higher SNR in this experiment.

7.2.3 Barker Code Sequence as Reference Signal. The input signal determines the excitation and thus strongly affects the estimation performance of the

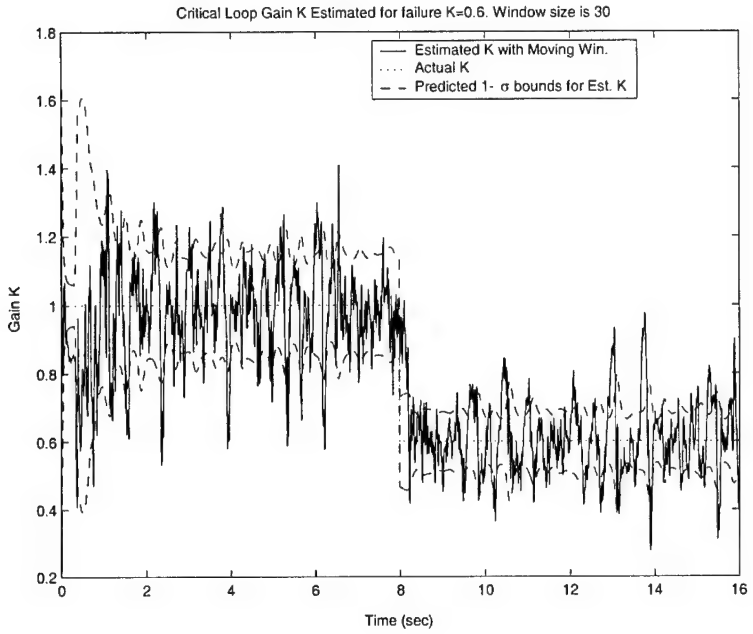


Figure 20. Loop gain K estimate provided by moving window identification algorithm with unit step ($1deg/sec$ pitch rate) input . Window size=30. $K_1 = 0.6$.

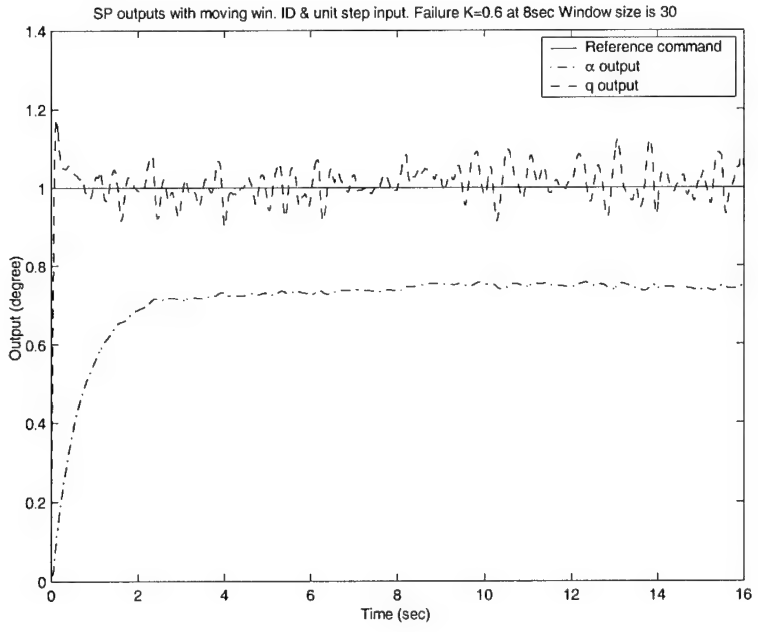


Figure 21. q and α response when unit step ($1deg/sec$ pitch rate) command is applied. Moving window system ID algorithm and fixed PI controller are used. $\sigma_\alpha = 0.03\,\mathrm{deg}$ and $\sigma_q = 0.1108\,\mathrm{deg}\,/\,\mathrm{sec}$. $K_1 = 0.6$.

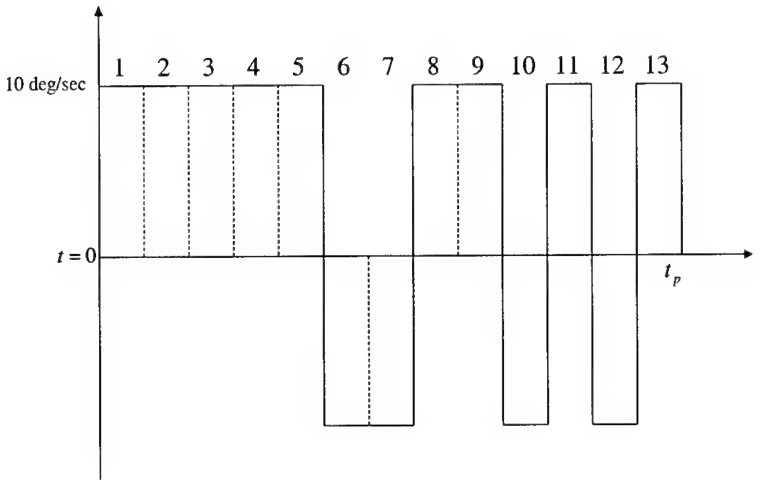


Figure 22. 13-bit Barker code.

system identification algorithm - as opposed to classical linear state estimation, i.e., Kalman filtering. The estimation benefits ensuing from using a strongly exciting 13-bit Barker code-like pilot command, as shown in Figure 22, are illustrated in this section. The Barker code is used in radar. Since there are three different "frequencies" in the 13-bit Barker code sequence, when it is used as the input command, we get strong excitation. In our simulation, each bit of the Barker code represents 1 second, and the total simulation time is now 13 seconds. The amplitude is 10 deg/sec. The failure is now simulated at $t_f = 6$ sec.

Fig. 23 shows the estimation performance of our moving-window system identification algorithm when the 13-bit Barker code is used as the reference signal. There are some spikes occurring during bits 1-5, 6-7, and 8-9, in the less excited period of the input sequence. After bit 10, the estimation performance is very good, since the input sequence is strongly exciting. Moreover, the estimation's settling time is very short.

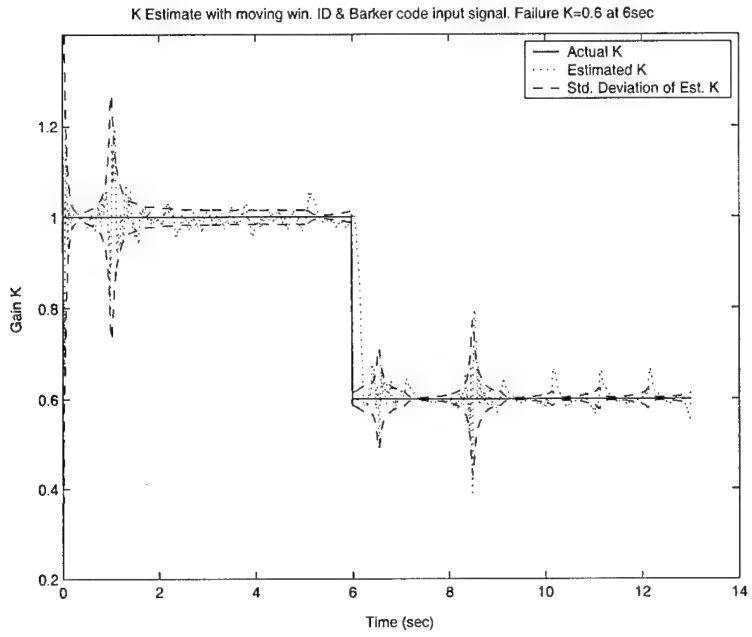


Figure 23. K estimate when moving window system ID algorithm, fixed PI Controller and 13-bit Barker input reference signal are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. Failure at $t_f = 6$ sec. and $K_1 = 0.6$.

The use of a strongly exciting 13-bit Barker code-like pilot command yields good estimation performance and it amply illustrates the effect of the input signal on system identification performance.

7.3 Smoothing Filter Performance

7.3.1 Fixed-Weights Smoother. The estimates of K obtained from the (short) moving-window system identification algorithm, with low excitation, and at relatively low SNRs, have a high σ_K, viz., they fluctuate. Hence, we use the fixed weights filter, Eq. (73) in Chapter 4, to smooth the parameter estimate before sending the latter to the PI controller module.

Smoothing of the system identification algorithm's parameter estimate helps to reduce the negative effects of noise, and consequently, bursting, a.k.a., poor estimation performance (during episodes of weak excitation) causing poor control performance. Thus, smoothing the system identification provided parameter estimate

improves the end-to-end tracking performance of the adaptive control system. Figure 24 shows the direct effect of smoothing on the loop gain estimate. The dashed lines represent a $\pm 20\%$ error bound about the true loop gain, K. We can see that the fixed-weights smoothing filter significantly reduces the fluctuation in the K estimate.

However, smoothing increases the identification time, which is of particular importance in a control surface failure scenario for an open-loop unstable aircraft. In Figure 25, the open-loop gain estimates shown are smoothed using various filter weights. As can be seen, when $\lambda = 0.7$ is used, the delay in failure detection time is greater than in the unsmoothed estimates case but is less than when the heavier smoothing weights $\lambda = 0.8$ and $\lambda = 0.9$ are used. A reduction in estimation delay comes at the expense of less smoothing. One must decide if this is an appropriate trade-off. Moreover, increasing the window length also helps to reduce the negative effects of noise and poor excitation, viz., the fluctuations in the open-loop gain estimate are reduced, but not as effectively as when a low-pass filter for the parameter estimate is used.

7.3.2 Adaptive Smoother. A fixed-weights smoother will reduce the fluctuations in $\widehat{K}$, but it will uniformly increase the identification delay, and, consequently, response time, of the identification algorithm, as is evident in Figure 25.

Figure 26 clearly shows the relation between the parameter estimate $\widehat{K}$ and the predicted standard deviation of the parameter estimation error, σ_K. The "spikes" in $\widehat{K}$ occur when σ_K is large. This indicates that our rigorously derived estimation algorithm indeed yields a reliable predicted parameter estimation error variance. Hence, we can confidently use the σ_K information. Moreover, Figure 27 shows the spikes in the parameter estimate to be strongly correlated with a sudden increase in the predicted parameter estimation error variance - in other words, whenever there is a spike in the $\sigma_K(k)/\sigma_K(k - w_l)$ ratio; w_l denotes the window length. We used

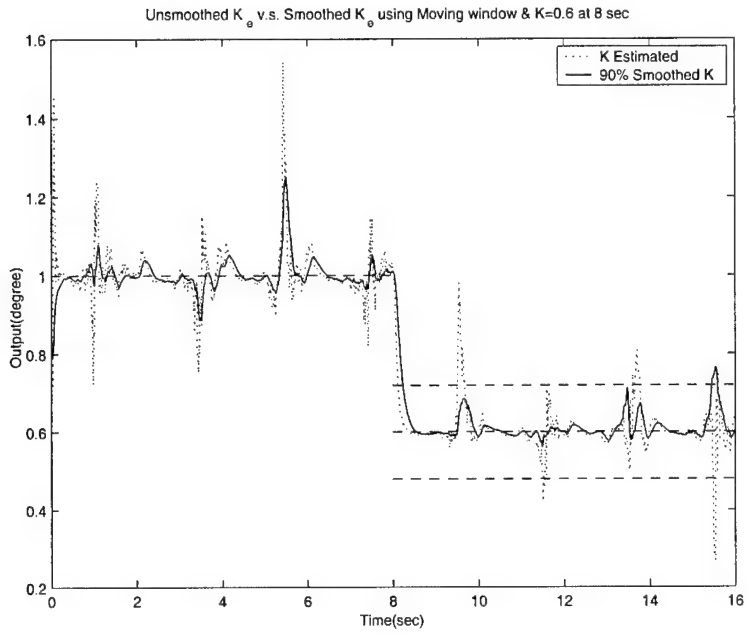


Figure 24. Nonsmoothed and 90% fixed weights smoothed loop gain estimate $\widehat{K}$ when moving window system ID algorithm is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

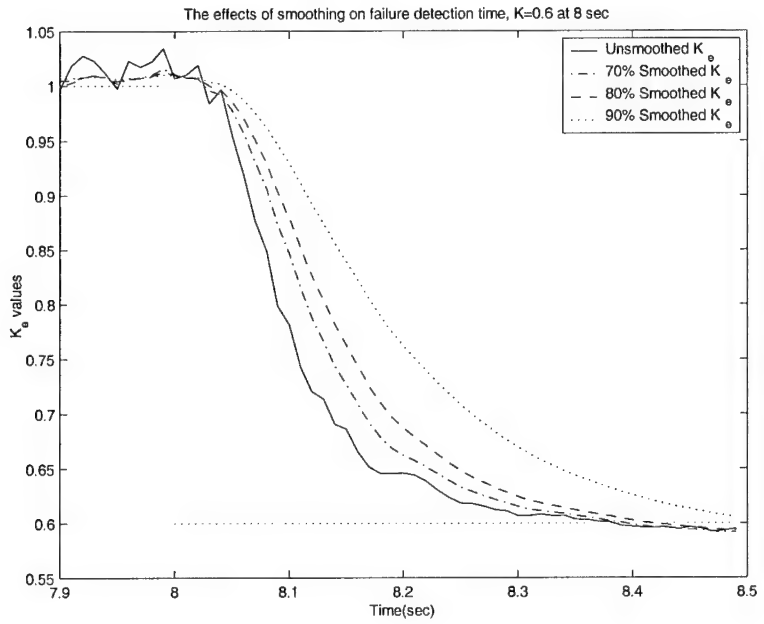


Figure 25. The effect of the fixed weights smoothing of estimated parameter on the settling time when moving window system ID is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

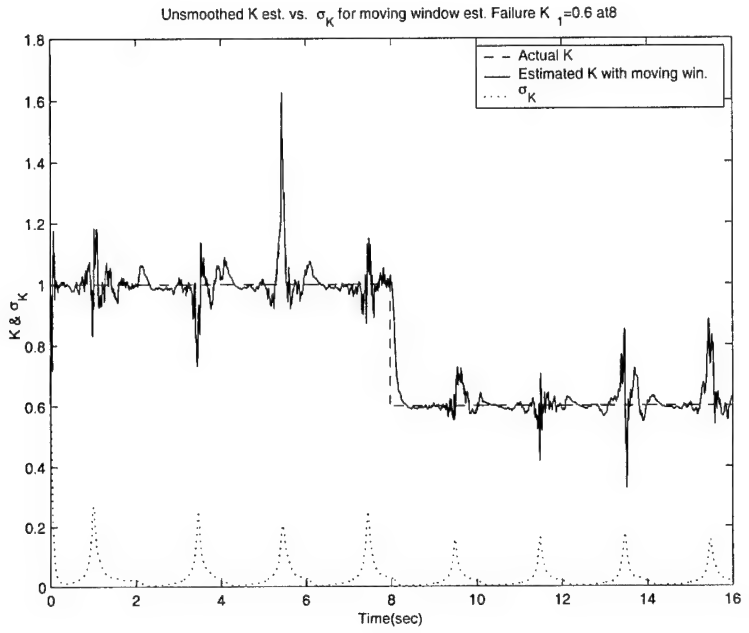


Figure 26. Loop gain estimate and predicted estimate error standard deviation when moving window system ID is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

Table 6. The Percentage of K Estimates Falling Into the 1-Sigma Range.

$\widehat{\mathbf{K}}$	$\widehat{\mathbf{K}}$ **Inside 1-σ**
Not smoothed	50.97%
70% Smoothed	52.64%
80% Smoothed	55.59%
90% Smoothed	61.02%
Adaptively Smoothed	65.90%

this insight to design an adaptive smoother, as specified in Eqs. (76) and (77) in Chapter 4.

Figure 28 shows the relation between the unsmoothed parameter estimate $\widehat{K}$, adaptively smoothed parameter estimate $\widehat{K}$ and the adaptively adjusted weight λ_k used in the adaptive parameter smoother. It indicates that $\lambda_k \approx 1$, i.e., use the previously smoothed estimate $\widehat{K}_{smooth}(k-1)$, when there is high fluctuation in estimated $\widehat{K}$. When the fluctuation is low, $\lambda_k \approx 0$, i.e., use the current estimate $\widehat{K}(k)$. The adaptive smoother did function as we expect and adaptively reduce the fluctuation in

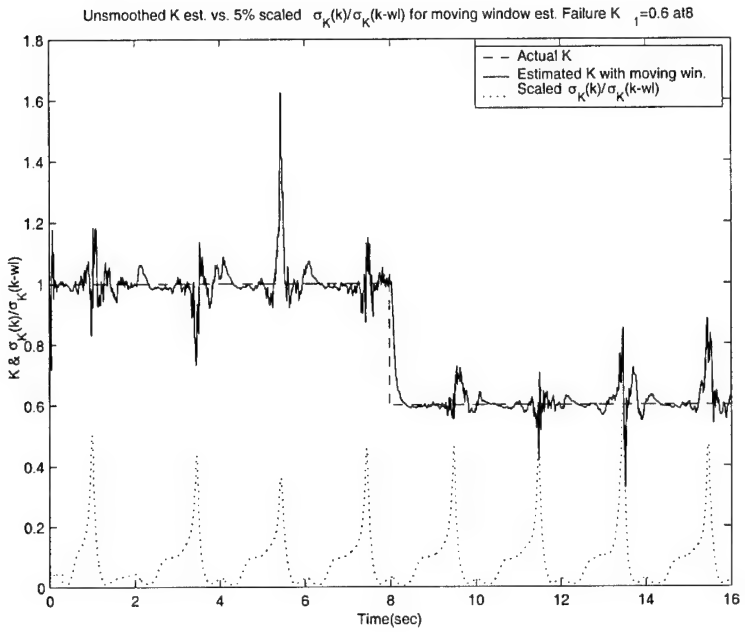


Figure 27. Loop gain estimate and $\sigma_K(k)/\sigma_K(k-w_l)$ when moving window system ID is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$. (where w_l denotes the window length)

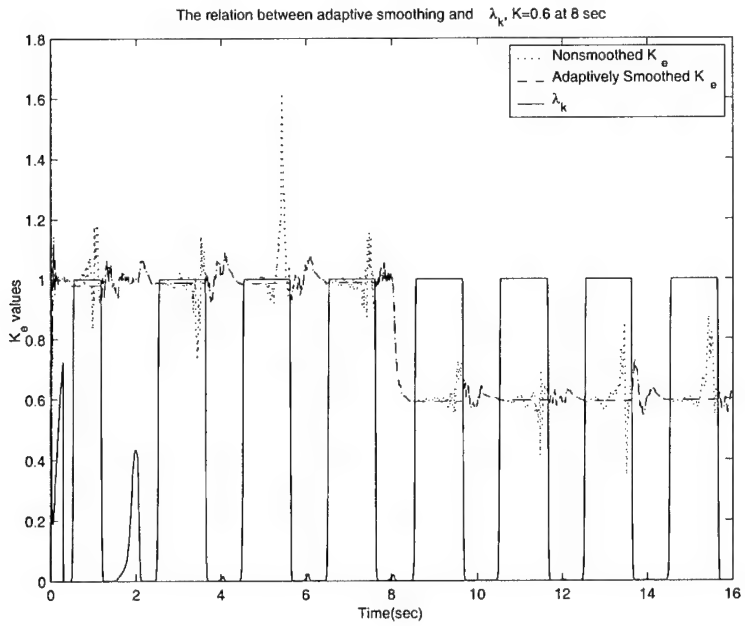


Figure 28. Loop gain estimate and λ_k when moving window system ID and adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

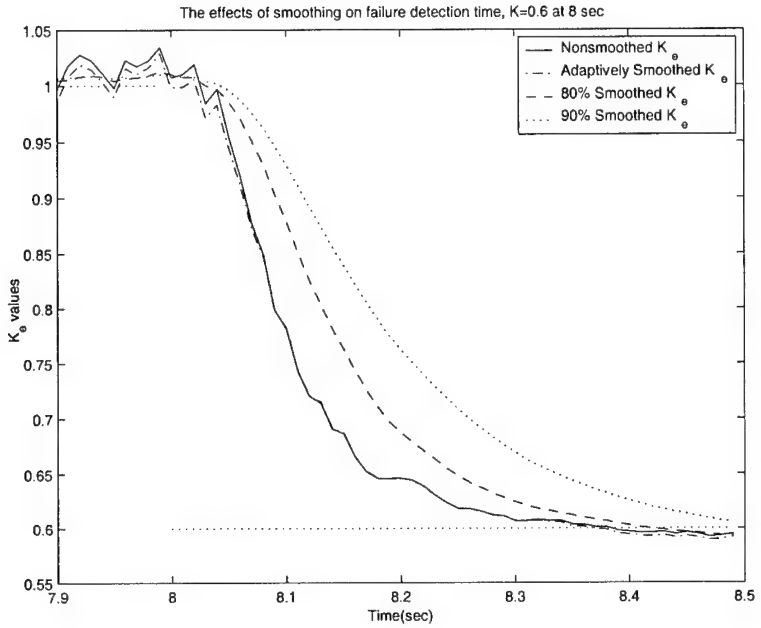


Figure 29. The effects of estimated parameter smoothing on the settling time when moving window system ID is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

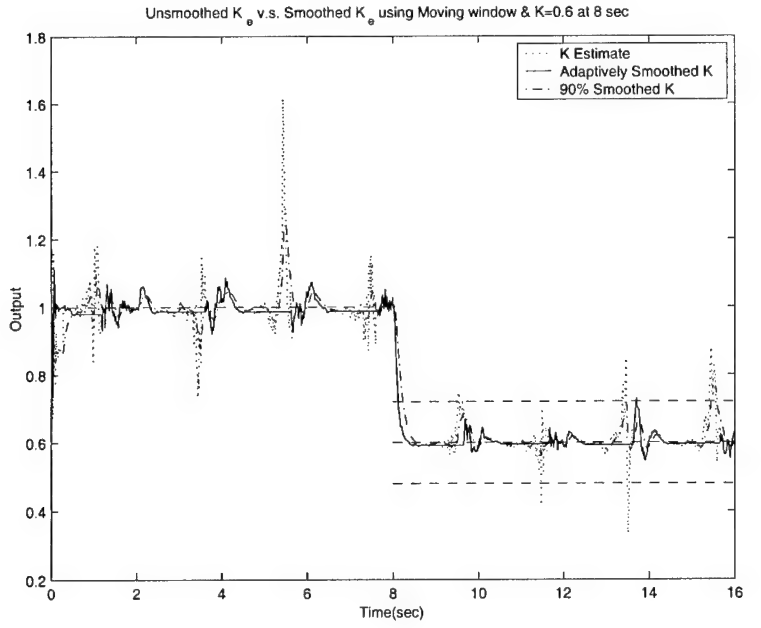


Figure 30. Loop gain estimate $\widehat{K}$ and smoothed $\widehat{K}_s$ when moving window system identification, fixed weights smoother and adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

parameter estimate. Figures 29 and 30 show the parameter estimation performance comparison with no smoother, a fixed-weights smoother, and an adaptive smoother for $\widehat{K}$, when the failure index is $K_1 = 0.6$ at $t_f = 8$ seconds. We can see that the adaptive smoother, Eq. (76), not only yields the fastest identification time, but also is more effective than a fixed-weights smoother. Table 6 shows the percentage of $\widehat{K}$ falling inside the predicted 1σ bounds when no smoother, a fixed-weights smoother and an adaptive smoother are used. We see that the adaptive smoother significantly "helps" the parameter estimation algorithm.

7.4 Tracking Controller Performance

In the tracking control experiments, the fixed linear PI tracking controller design in Chapter 5 is exercised first. This is a fairly robust controller and it yields a solid benchmark against which the adaptive and reconfigurable controller's performance is gauged. Next, the tracking performance of our adaptive and reconfigurable control system is evaluated. The tracking performance when using the "exciting" 13-bit Barker code sequence pilot command is also discussed.

7.4.1 Fixed Proportional Controller Performance. In general, post-failure tracking performance of the fixed proportional controller for the various reduced open-loop gain values is similar to that of the PI controller. The post-failure tracking performance of the proportional controller becomes unacceptable when the critical 1oop gain is reduced to $K_1 = 0.2$. When the critical loop gain is further reduced, tracking performance deteriorates rapidly as shown in Figures 31 to 34.

7.4.2 Fixed PI Controller Performance. Our fixed PI tracking controller developed in Chapter 5 is, by design, fairly robust. In the simulation experiments, at time $t = 8$ sec. into the flight, the open-loop gain K is reduced to $K_1 = 0.8$, $K_1 = 0.6$, $K_1 = 0.2$, $K_1 = 0.1$ and $K_1 = 0.06$. Although no discernible loss in post-failure tracking performance is recorded for $K_1 = 0.8$ in Figure 35 (due to the

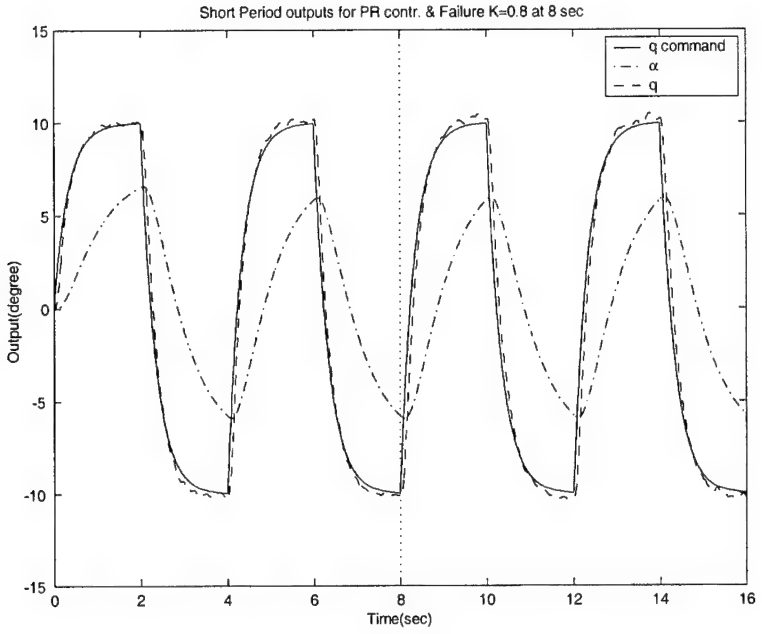


Figure 31. Pitch rate q and angle of attack α responses when the fixed proportional controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.8$.

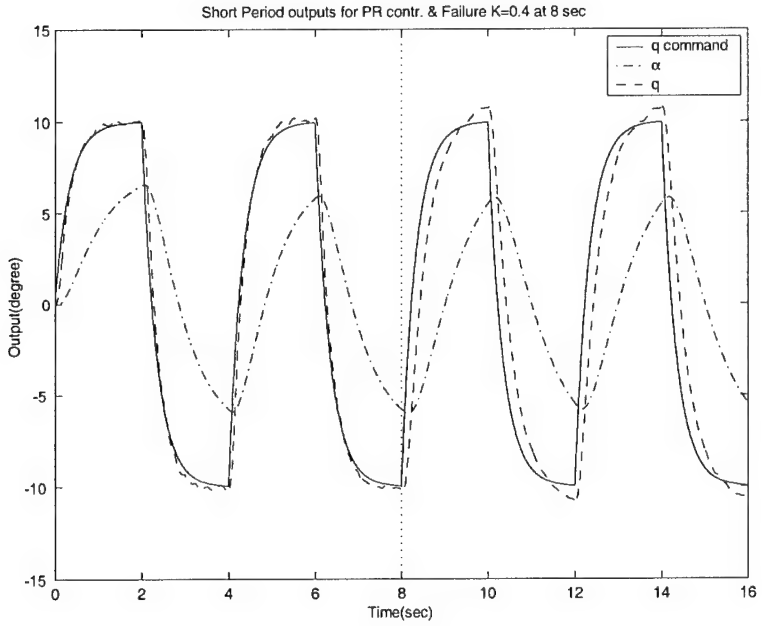


Figure 32. Pitch rate q and angle of attack α responses when the fixed proportional controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.4$.

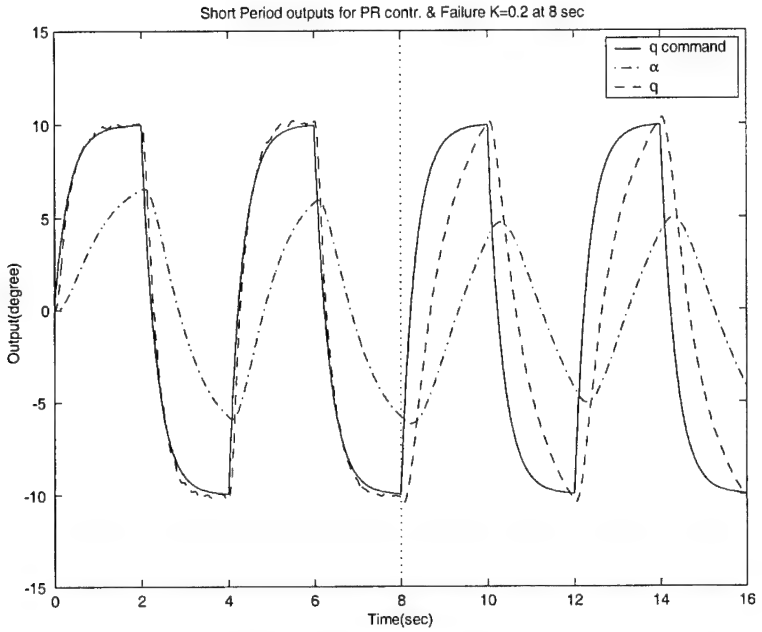


Figure 33. Pitch rate q and angle of attack α responses when the fixed proportional controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.2$.

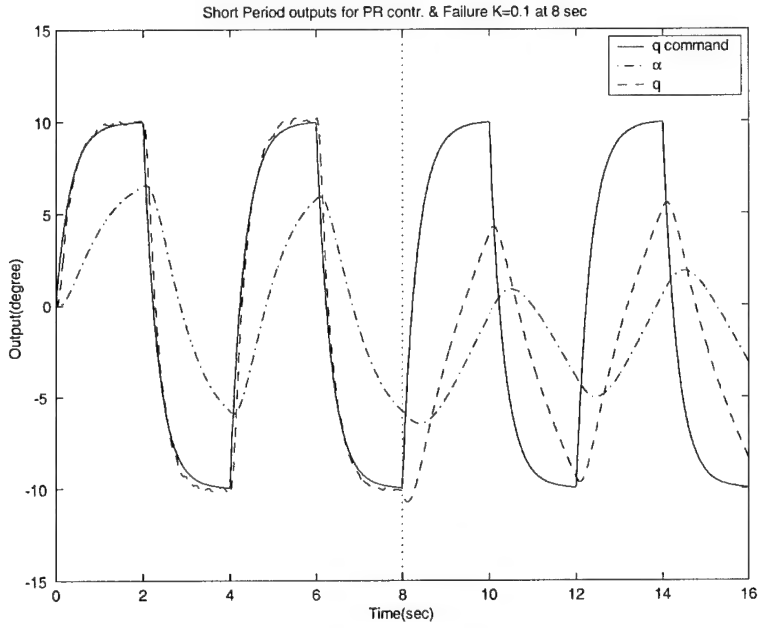


Figure 34. Pitch rate q and angle of attack α responses when the fixed proportional controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.1$.

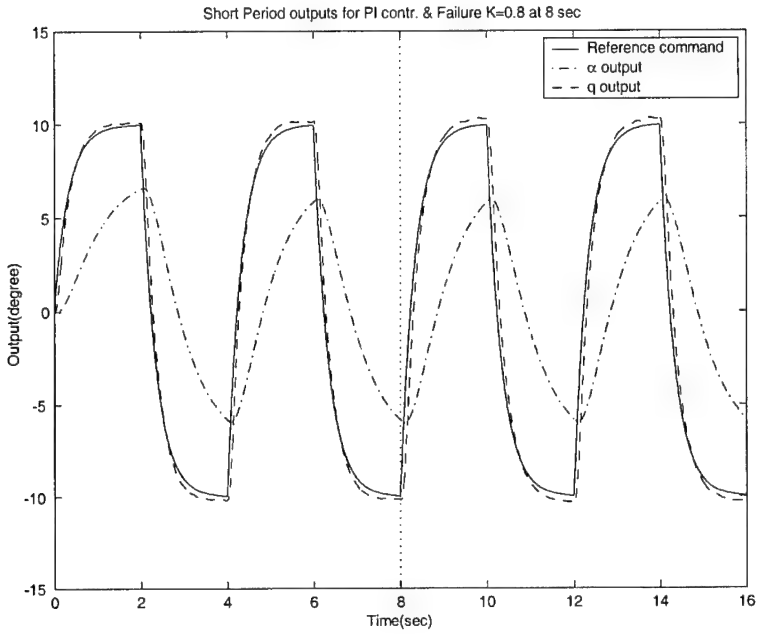


Figure 35. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.8$.

robustness of the fixed PI controller), the degradation in tracking performance is pronounced for $K_1 = 0.2$, as seen in Figure 38. Although the control system does not become unstable until a degree of failure which corresponds to $K_1 = 0.08$ - see, e.g., Table 3 - the post-failure tracking performance of the fixed PI controller falls out of acceptable limits before this point, approximately when the loop gain $K_1 \leq 0.2$ - see, e.g. Figure 38, where the results for $K_1 = 0.2$ are shown. When the K value further decreases to $K_1 = 0.1$ and below, as shown in Figures 39 and 40, post-failure tracking performance of the fixed PI controller deteriorates significantly and is not acceptable. In both cases, when either the fixed PI or the proportional tracking controller were used, very similar identification results were obtained.

7.4.3 Adaptive and Reconfigurable Control.

7.4.3.1 Expanding Window System Identification. Now, a "conventional", two-module, adaptive and reconfigurable controller is implemented. The

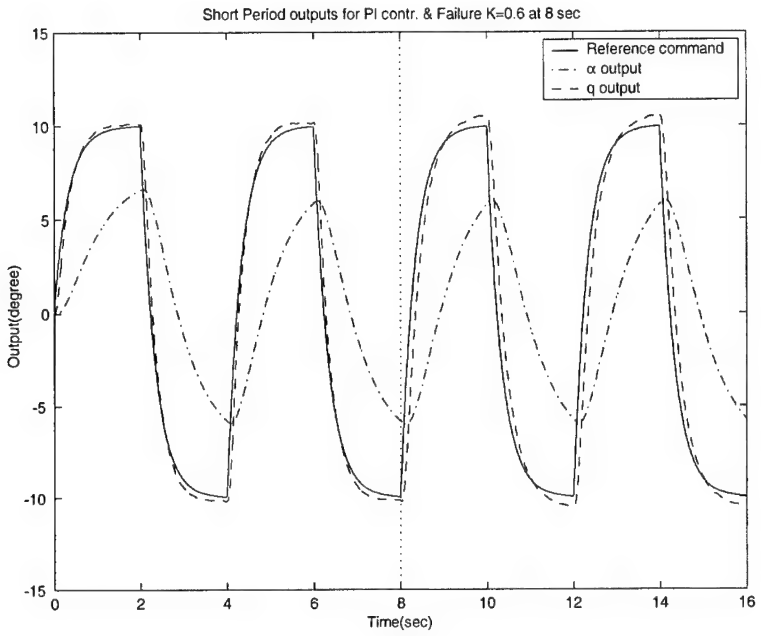


Figure 36. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

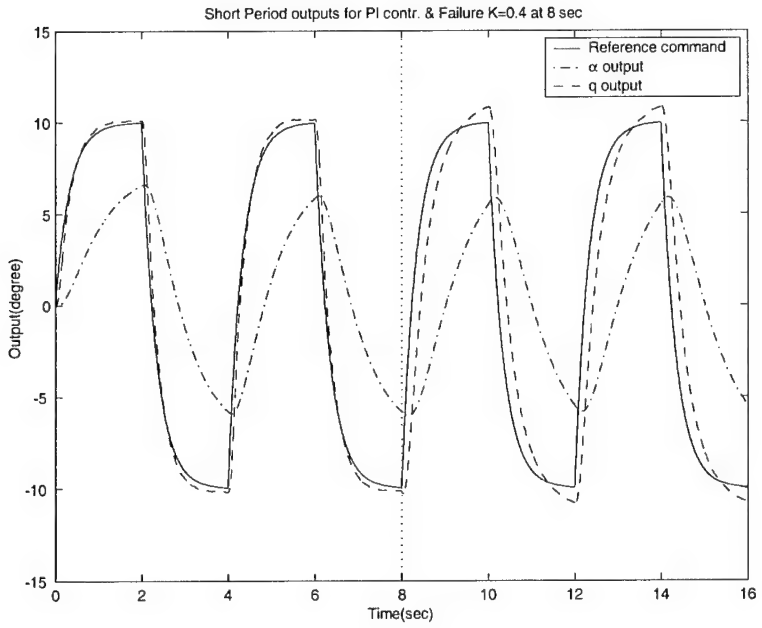


Figure 37. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.4$.

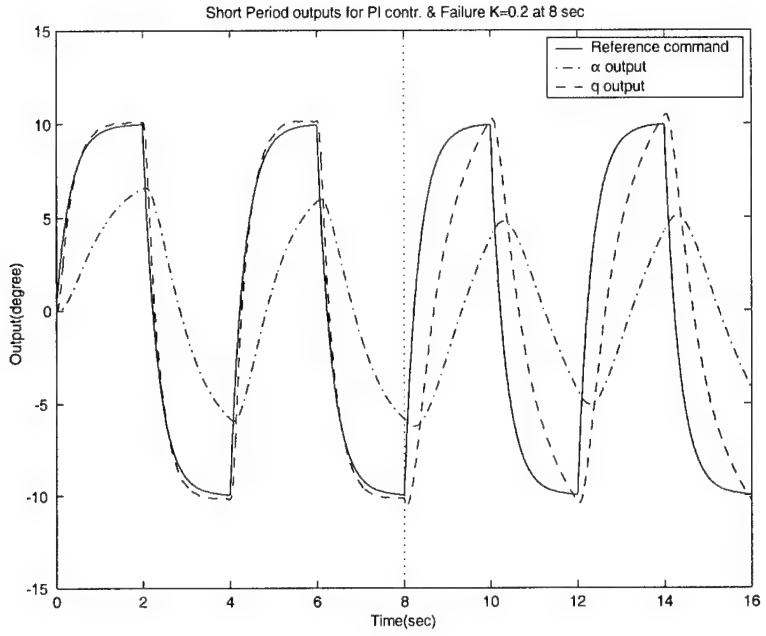


Figure 38. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg / sec}$. $K_1 = 0.2$.

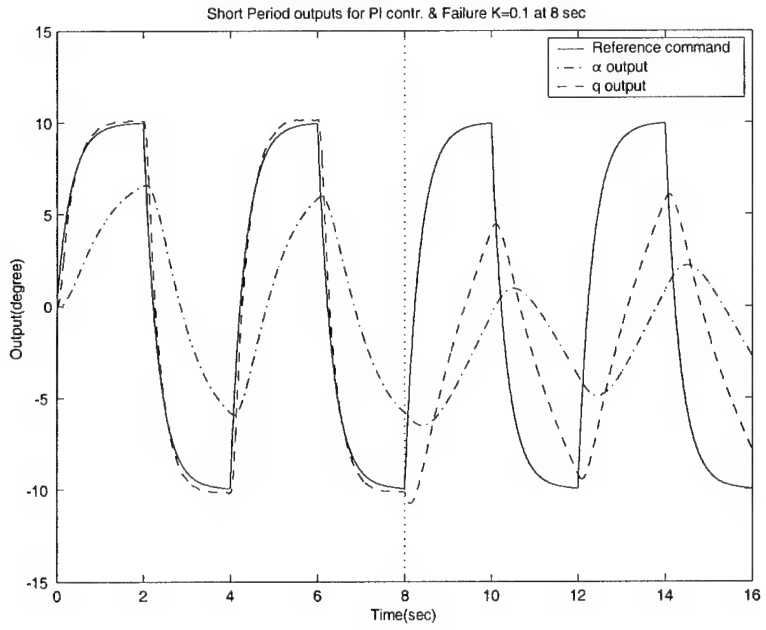


Figure 39. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg / sec}$. $K_1 = 0.1$.

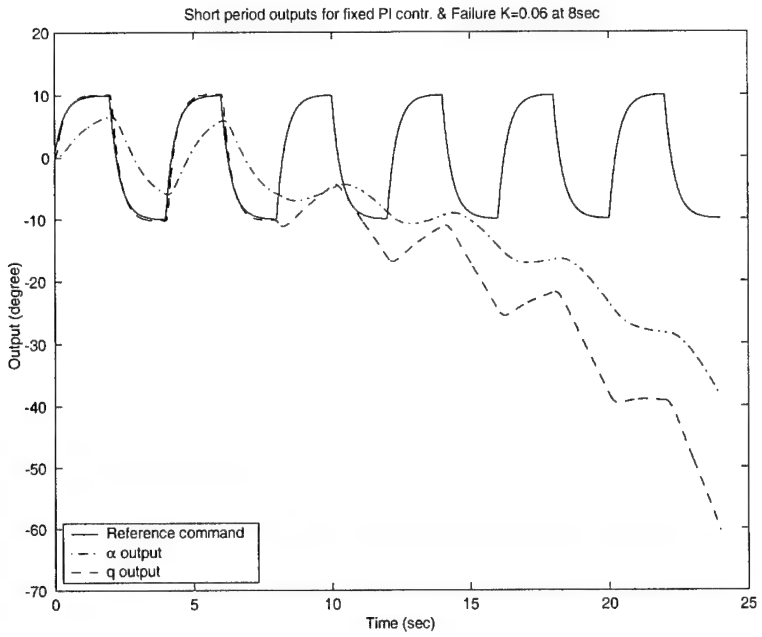


Figure 40. Pitch rate q and angle of attack α responses when the fixed PI controller is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.06$.

controller entails an expanding window system identification module and a variable gain controller. When the expanding window-based system identification algorithm is used to estimate the states α, q, and the open-loop gain parameter (K) for degrees of failure of $K_1 = 0.8$, $K_1 = 0.6$, $K_1 = 0.2$, $K_1 = 0.1$ and $K_1 = 0.06$, and no parameter estimate smoothing filter is used, the tracking performance is shown in Figures 41 - 45. After the point of failure, a considerable error between the estimated pitch rate and the commanded pitch rate develops. This is mainly due to the estimation lag in the expanding window system identification module. However, the tracking performance improves as time passes, and the expanding window system identification algorithm settles on a good parameter estimate, and the pitch rate then tracks the commanded pitch rate. When the open-loop gain dropped to $K_1 = 0.06$ in Figure 42, tracking at $t \approx 10$ second is temporarily poor, because the parameter identification delay is long, and the fixed PI tracking controller is only able to tolerate a failure of $K_1 = 0.08$ before the control system becomes unstable. However, after a temporary lapse, the adaptive and reconfigurable controller affords a recovery at $t \approx 16$ second.

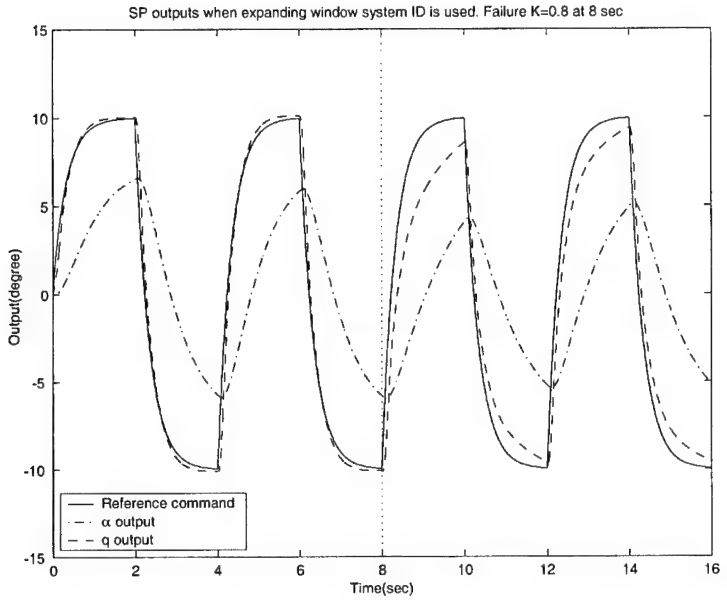


Figure 41. q and α responses when expanding window system identification algorithm and reconfigurable control are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.8$.

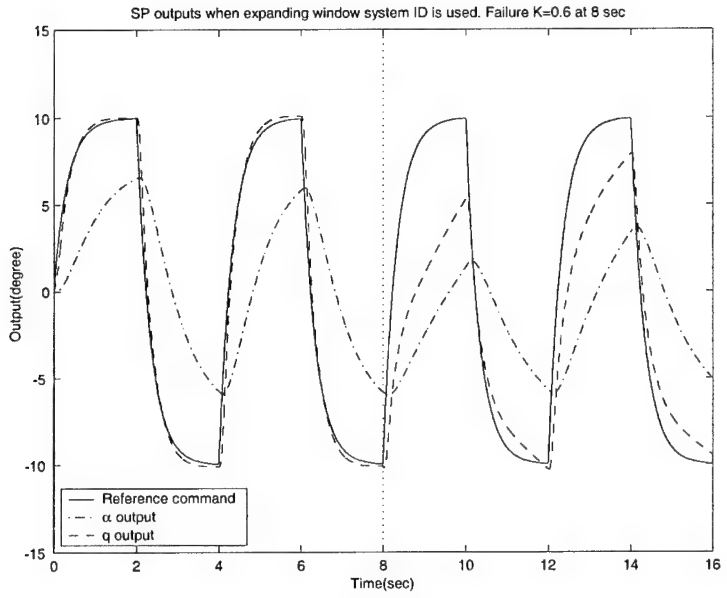


Figure 42. q and α responses when expanding window system identification algorithm and reconfigurable control are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

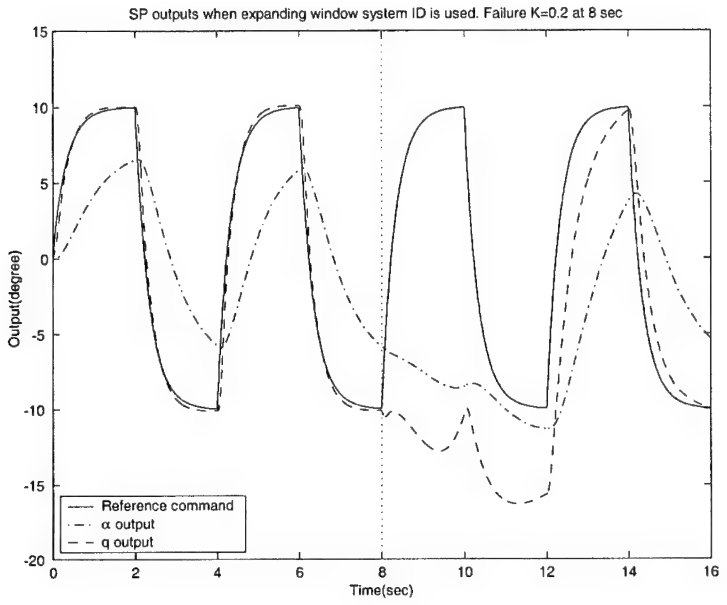


Figure 43. q and α responses when expanding window system identification algorithm and reconfigurable control are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.2$.

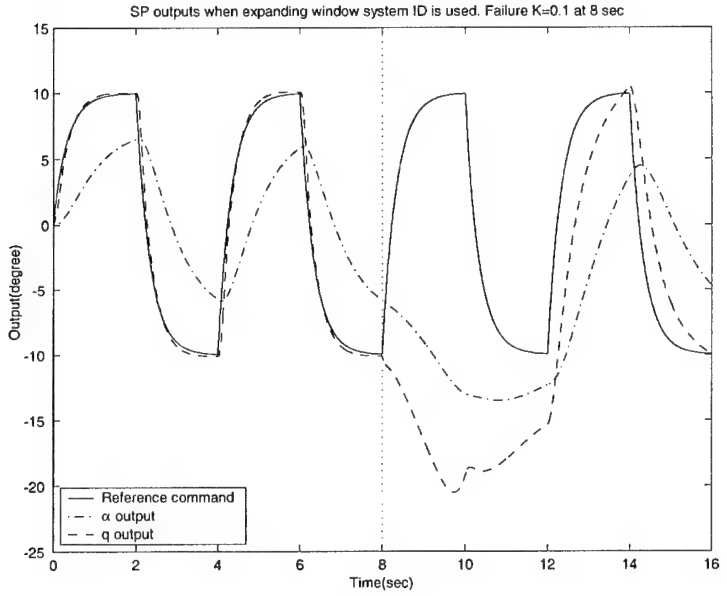


Figure 44. q and α responses when expanding window system identification algorithm and reconfigurable control are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.1$.

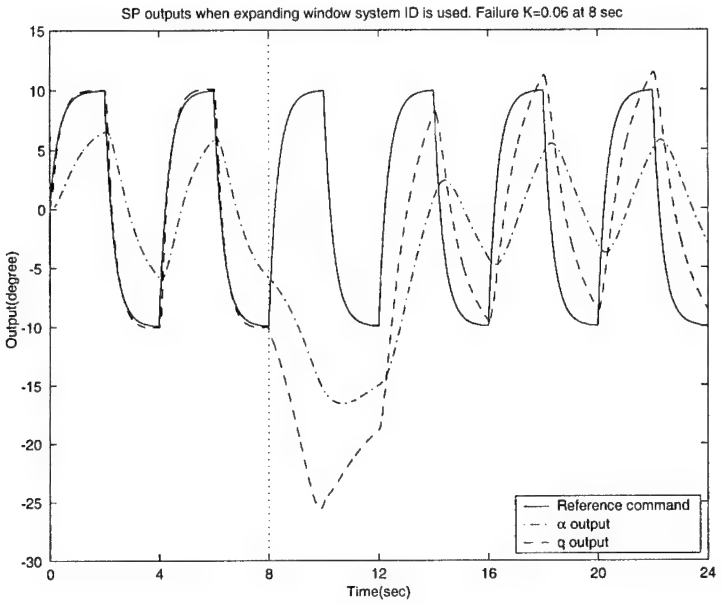


Figure 45. q and α responses when expanding window system identification algorithm and reconfigurable control are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.06$.

7.4.3.2 Moving-Window System Identification. The tracking performance of the complete, three-module, adaptive and reconfigurable controller, using the moving-window system identification algorithm, is shown in Figures 46 - 49. After the failure, and when $K_1 = 0.8$, only a small tracking error occurs between the commanded pitch rate and the pitch rate output. As the degree of failure increases, the tracking error increases too. However, the tracking performance is much better than that of the fixed PI tracking controller, and the previously discussed two-module adaptive tracking controller using an expanding window-based system identification algorithm. Concerning the latter, the estimation lag is now reduced. This is evident when we compare Figures 45 and 49, where the open-loop gain dropped to $K_1 = 0.06$. In Figure 45 where the estimation lag was high, tracking immediately after the failure was no longer acceptable because the fixed PI tracking controller is

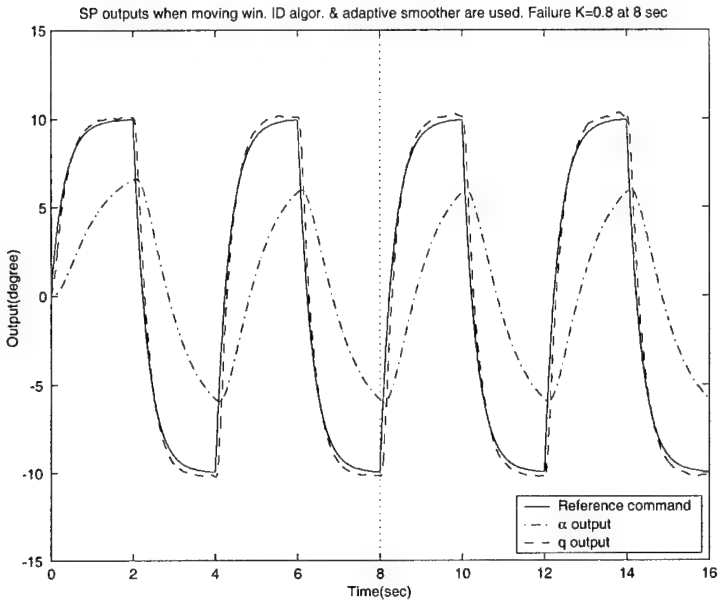


Figure 46. q and α response when moving window system ID algorithm & adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.8$.

only able to tolerate a failure of $K_1 = 0.08$ before letting the control system become unstable.

7.4.3.3 13-bit Barker Code Command. Figure 50 shows the tracking performance of the fixed PI controller when the dynamic 13-bit Barker code sequence is used as reference command. Figures 51 and 52 illustrate the tracking performance of our two and three-module adaptive and reconfigurable controller using the expanding window system identification algorithm and moving-window system identification algorithm, respectively, when the dynamic 13-bit Barker code sequence is used as reference command, and the control surface loss is $K_1 = 0.6$ and it occurs at $t_f = 6$ seconds. The tracking performance of the moving-window system identification algorithm is superior to that of the expanding window system identification algorithm and is slightly better than the fixed PI controller's; recall, however, that the fixed PI controller cannot handle a severe failure, e.g., $K_1 = 0.2$.

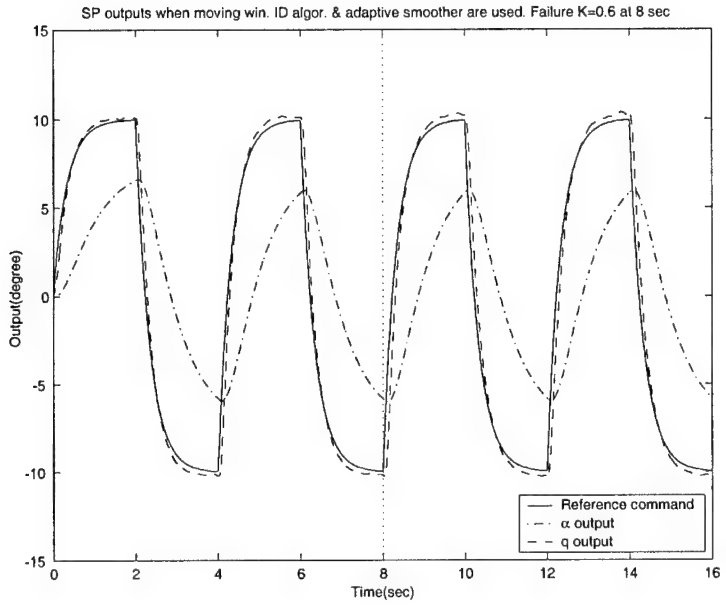


Figure 47. q and α response when moving window system ID algorithm & adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.

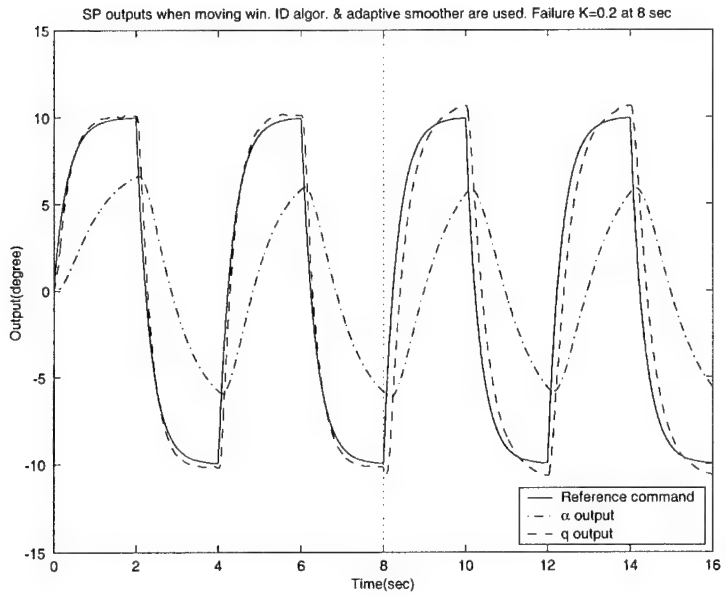


Figure 48. q and α response when moving window system ID algorithm & adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.2$.

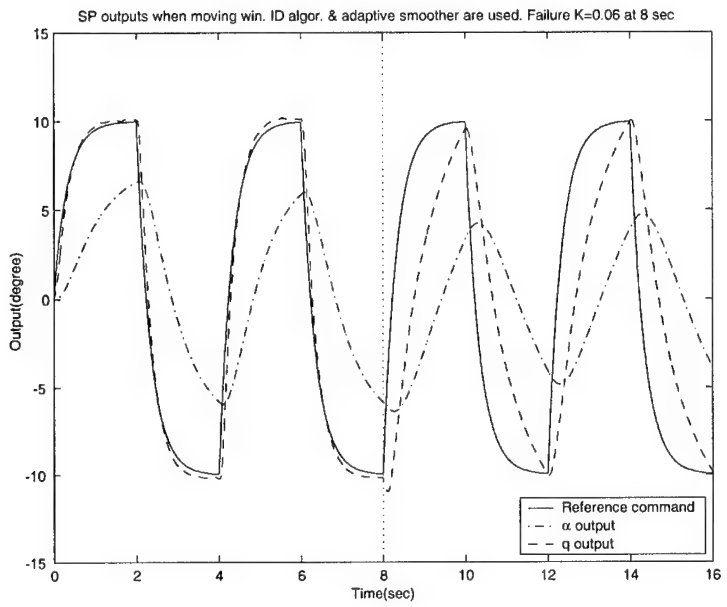


Figure 49. q and α response when moving window system ID algorithm & adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.06$.

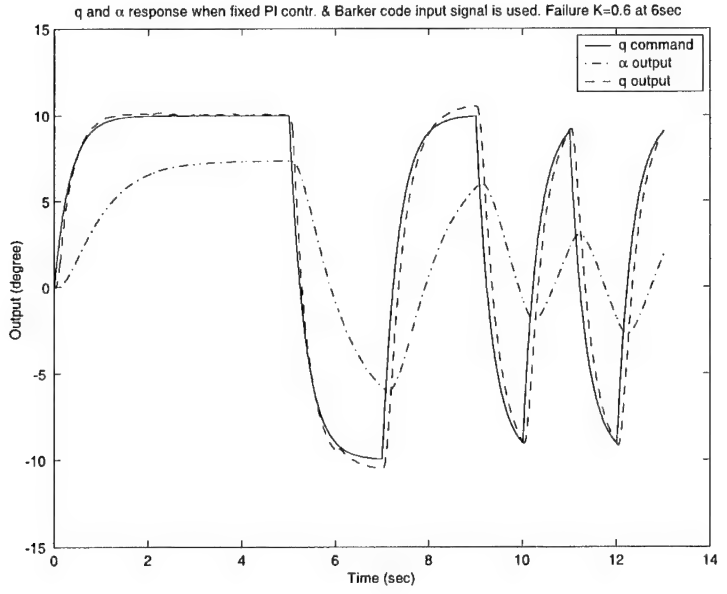


Figure 50. q and α response when fixed PI controller & 13-bit Barker code input signal are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. Failure at t=6 sec. and $K_1 = 0.6$.

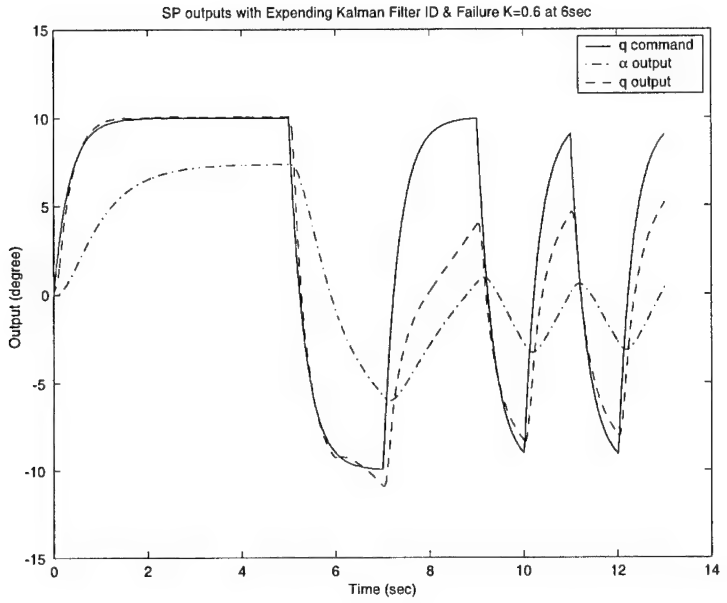


Figure 51. q and α response when expanding window system identification algorithm & 13-bit Barker code input signal are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. Failure at t=6 sec. and $K_1 = 0.6$.

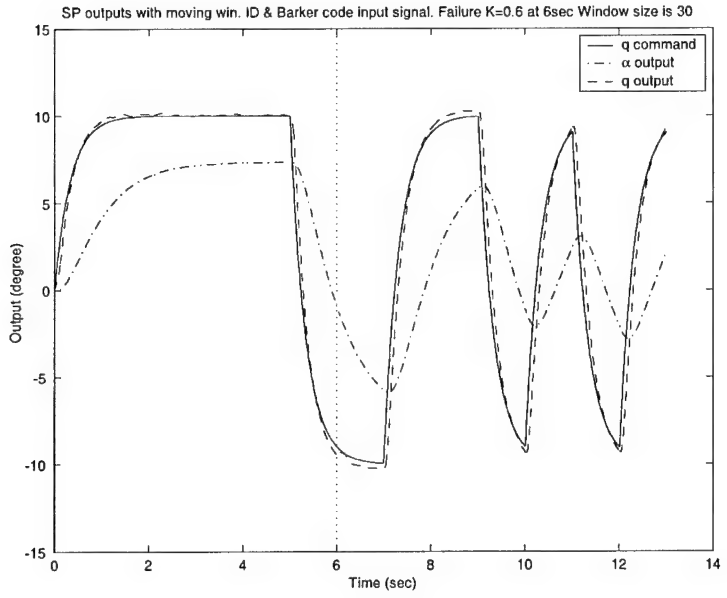


Figure 52. q and α response when moving window system ID algorithm & 13-bit Barker code input signal are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. Failure at t=6 sec. and $K_1 = 0.6$.

7.5 SNR Effects

With the moving-window system identification algorithm, the accuracy of the results strongly depends on the measurement's Signal-to-Noise Ratio (SNR). This is due to the short data window used to calculate the estimates. In the previous section, a SNR of $38.9557dB$ ($\sigma_\alpha = 0.03$ [deg] and $\sigma_q = 0.1108$ [deg/sec]) was used. In this section several additional SNRs listed in Table 4 are experimented with in order to analyze the $SNR's$ effect on the estimation and tracking performance.

Figures 53 to 57 show that, as the measurement's Signal-to-Noise Ratio increases, the accuracy of the parameter estimate increases, and the tracking performance also increases. The fluctuations in the gain estimate, and bursting, caused by instances of poor excitation, is significantly reduced at high $SNRs$.

7.6 Window Size Effects

The effect of the length of the data window on the estimation and tracking performance of the system identification algorithm are investigated. We set the window sizes w_l used in the moving-window system identification algorithm at 50, 40, 30, 20, and 10, the post-failure open-loop gain at $K_1 = 0.6$, and the Signal-to-Noise Ratio of the measurement is $SNR = 38.9557dB$.

Figures 58 to 61, and Figure 54 show that a larger window size yields better parameter estimates and consequently better tracking performance. Using longer windows reduces the fluctuations in the parameter estimate, and bursting. However, as shown in Figure 62, longer windows bring about a delay in the estimation of the loop gain after the failure.

7.7 Unmodeled Dynamics Effects

7.7.1 Unmodeled Phugoid Dynamics. When we add the Phugoid dynamics to our short period A/C model with the first-order actuator augmented dynamics -

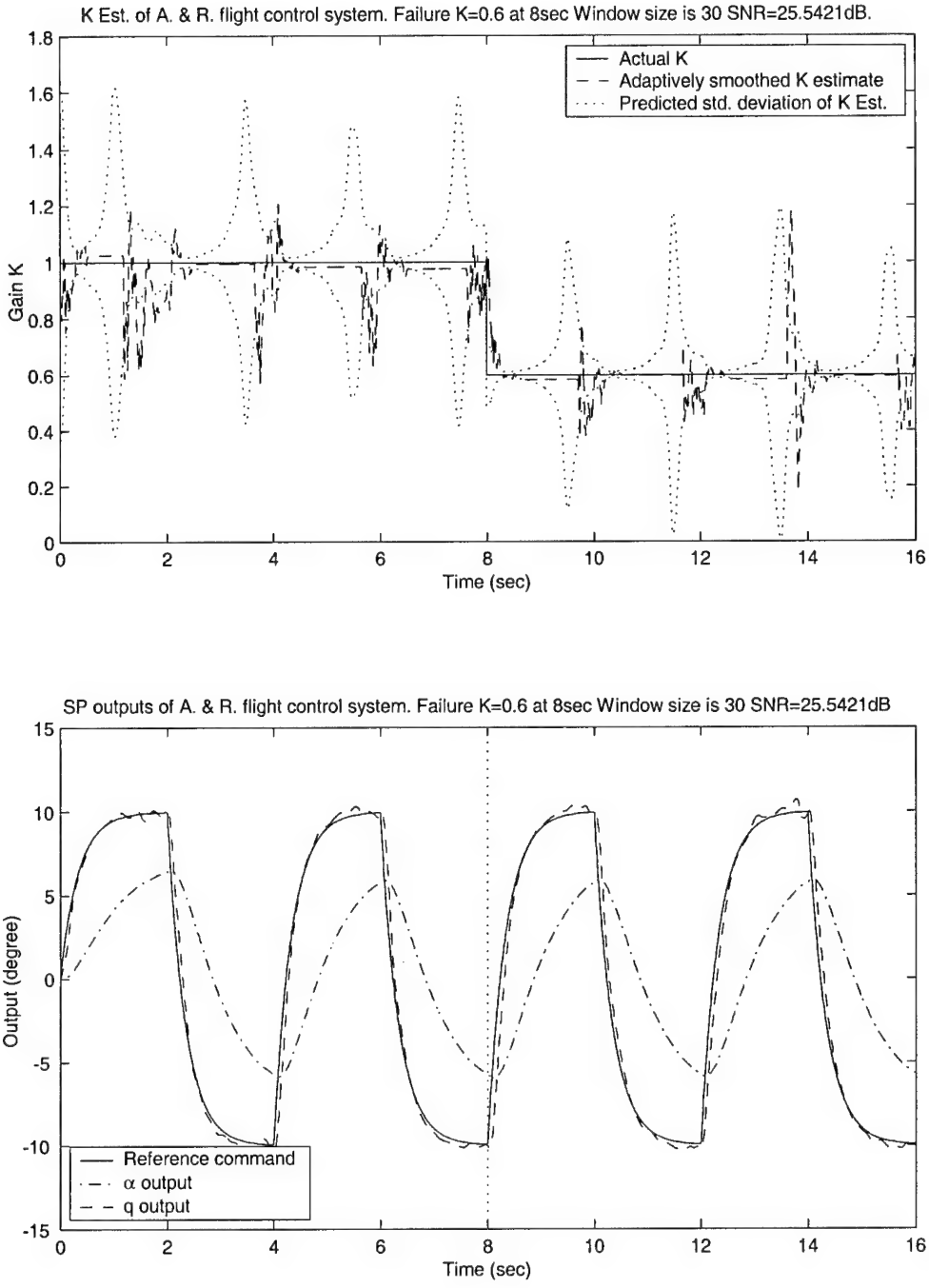


Figure 53. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. $SNR = 25.5421dB$ ($\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.55534\,\text{deg}/\,\text{sec}$). Failure at t=8 sec. $K_1 = 0.6$.

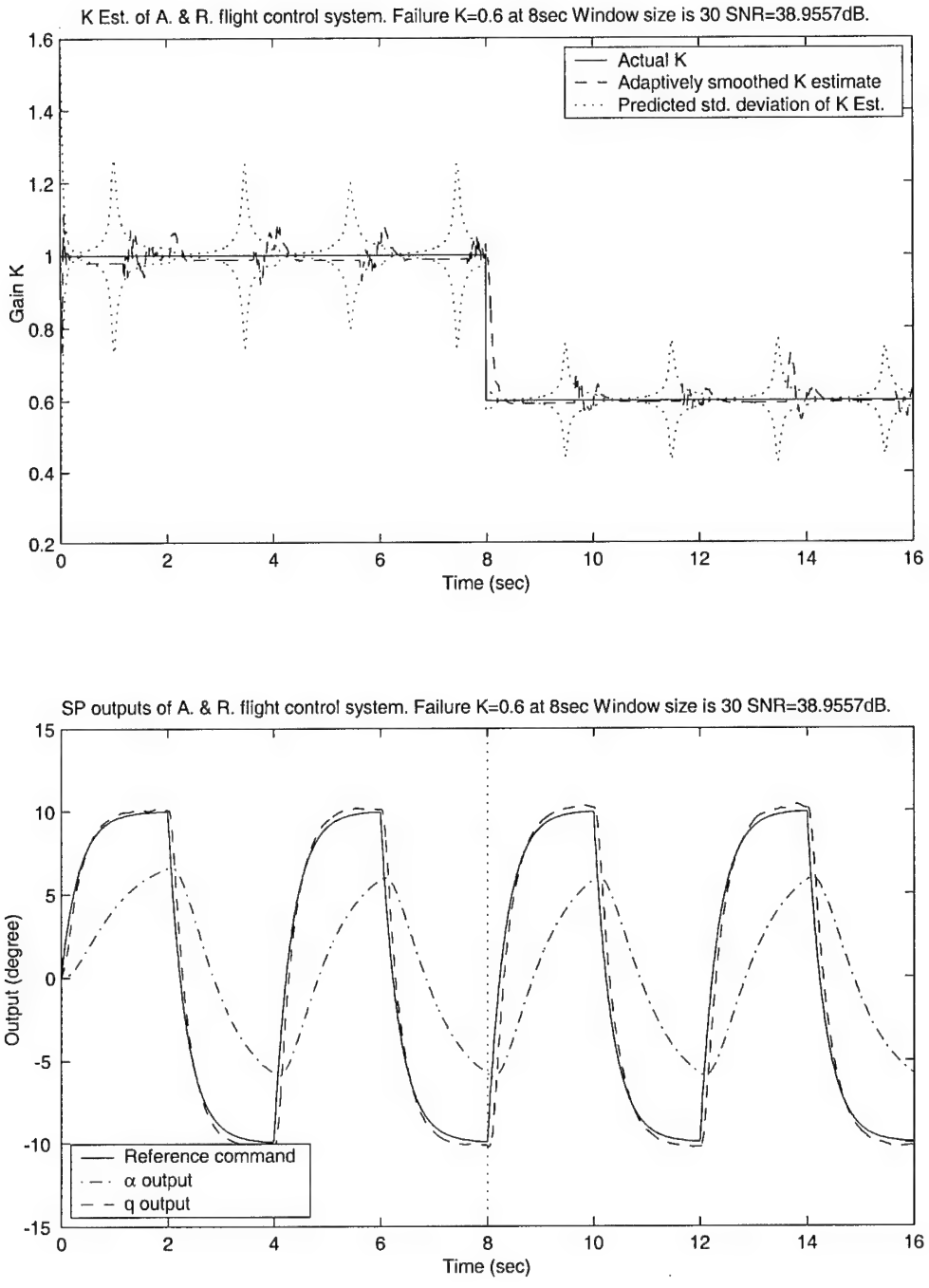


Figure 54. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. $SNR = 38.9557dB$ ($\sigma_\alpha = 0.03$ deg and $\sigma_q = 0.1108$ deg / sec). Failure at t=8 sec. $K_1 = 0.6$.

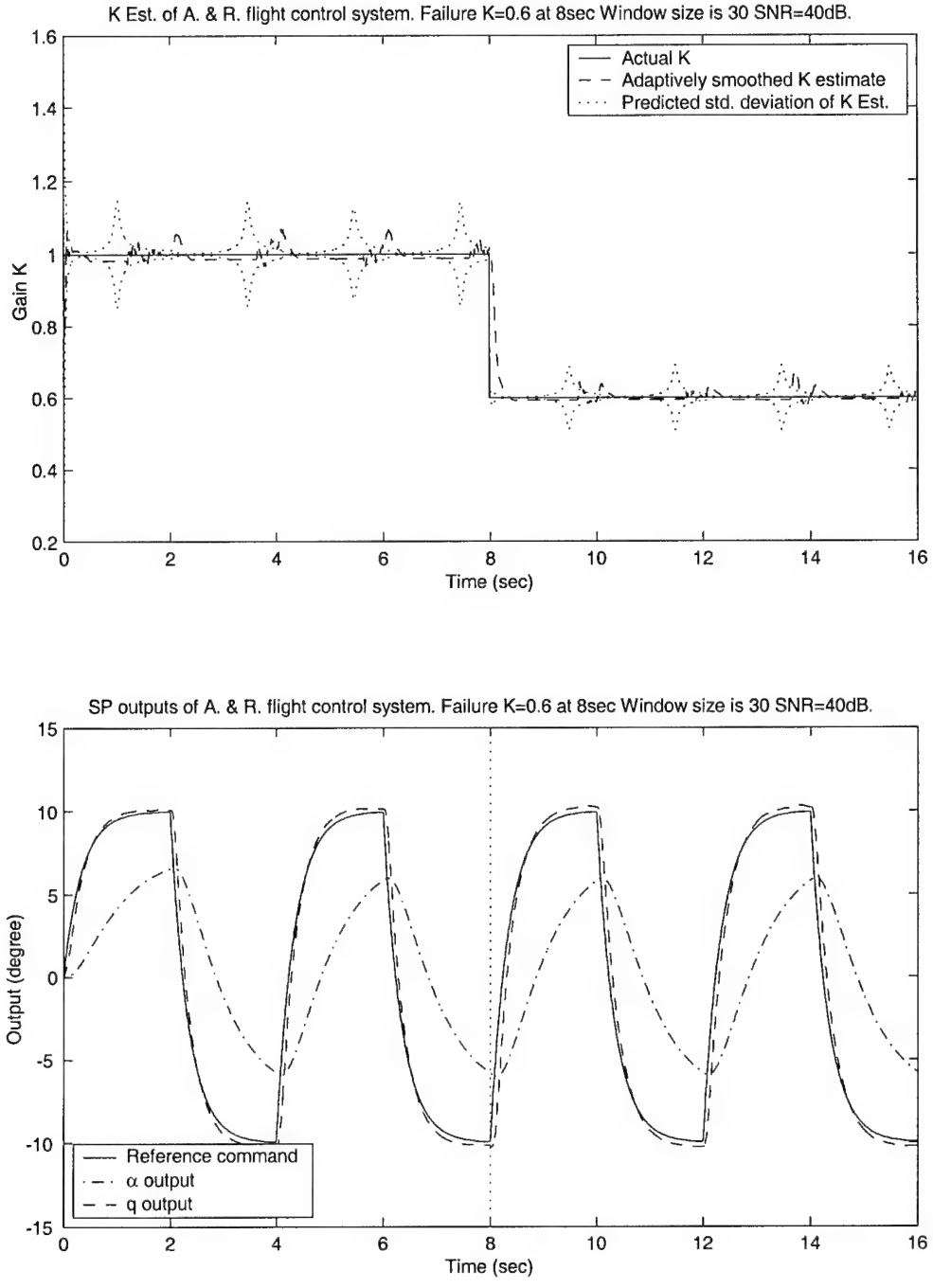


Figure 55. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. $SNR = 40dB$ ($\sigma_\alpha = 0.0602\,\text{deg}$ and $\sigma_q = 0.0602\,\text{deg}\,/\,\text{sec}$). Failure at t=8 sec. $K_1 = 0.6$.

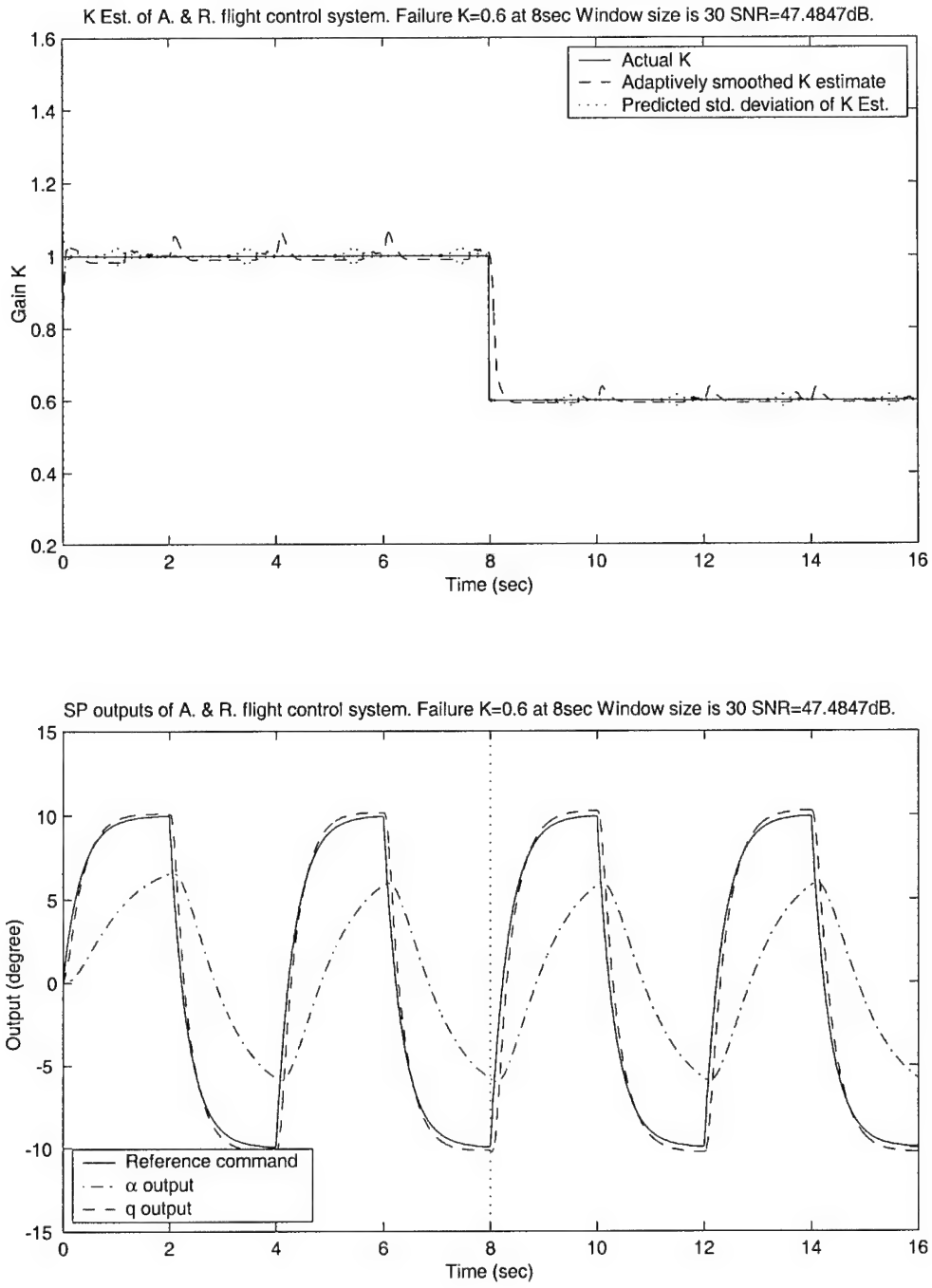


Figure 56. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. $SNR = 47.4847dB$ ($\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.01108\,\text{deg}/\,\text{sec}$). Failure at t=8 sec. $K_1 = 0.6$.

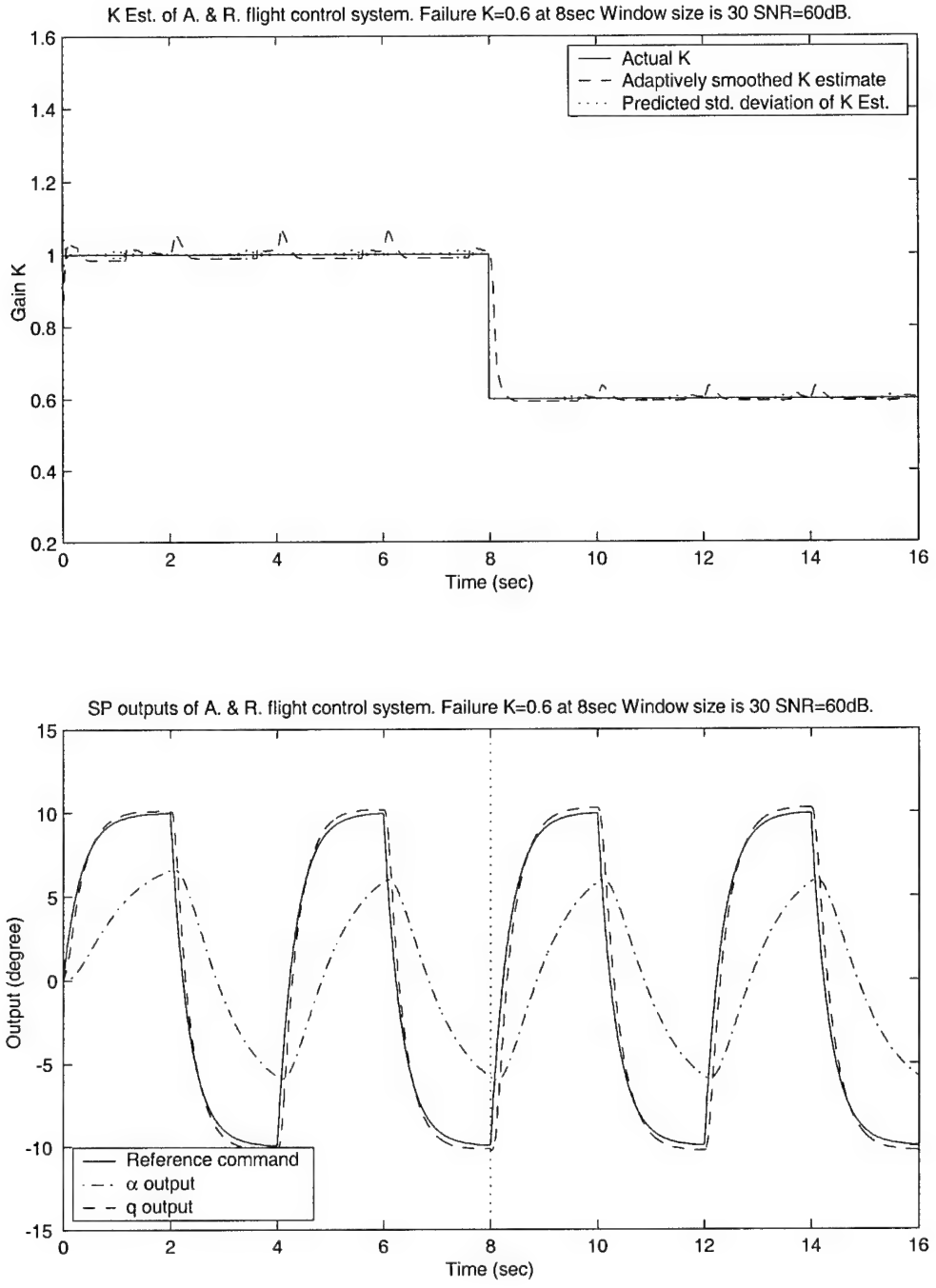


Figure 57. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. $SNR = 60dB$ ($\sigma_\alpha = 0.00602\,\text{deg}$ and $\sigma_q = 0.00602\,\text{deg}\,/\,\text{sec}$). Failure at t=8 sec. $K_1 = 0.6$.

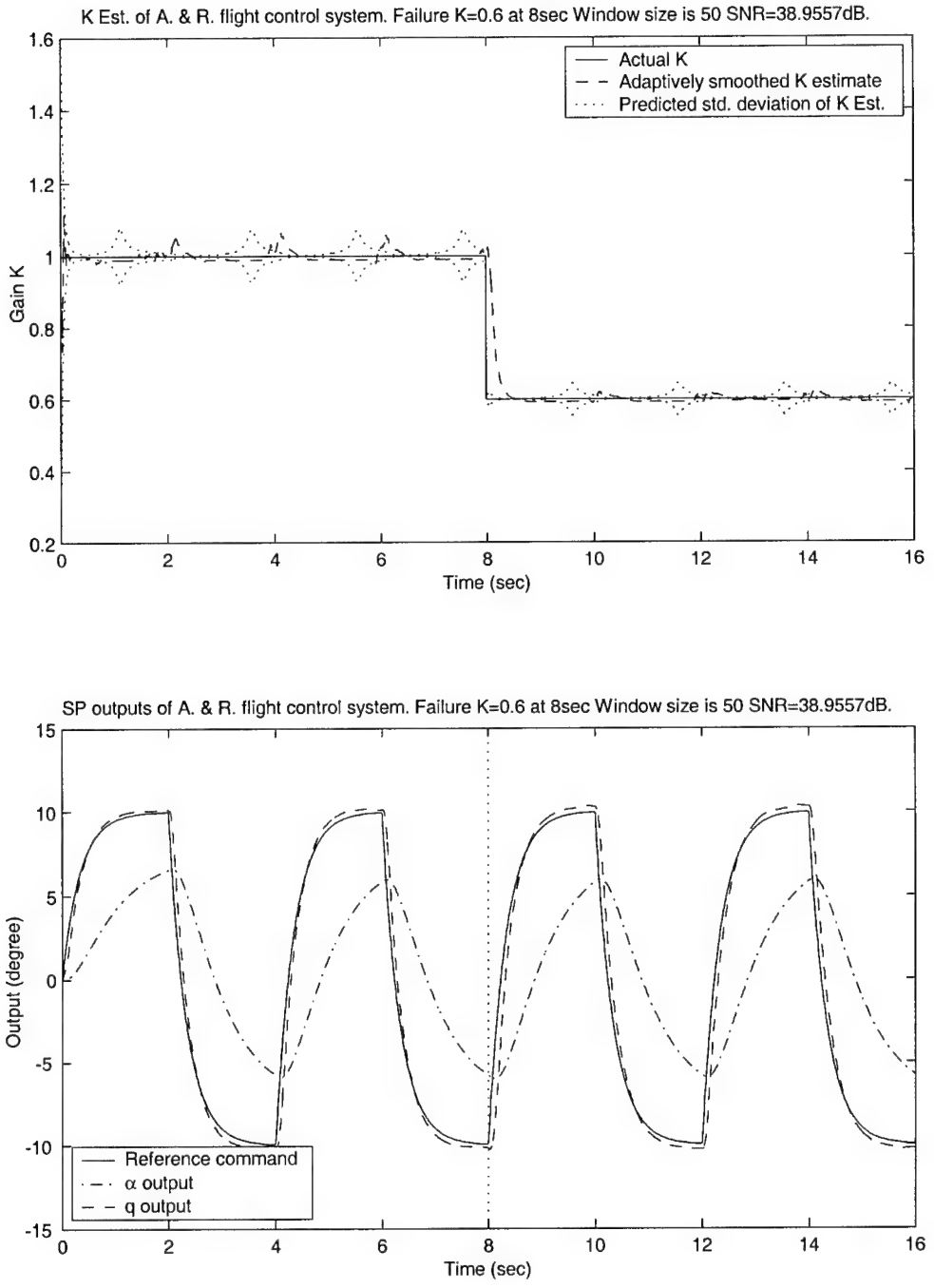


Figure 58. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. Window size is 50, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. Failure at t=8 sec. $K_1 = 0.6$.

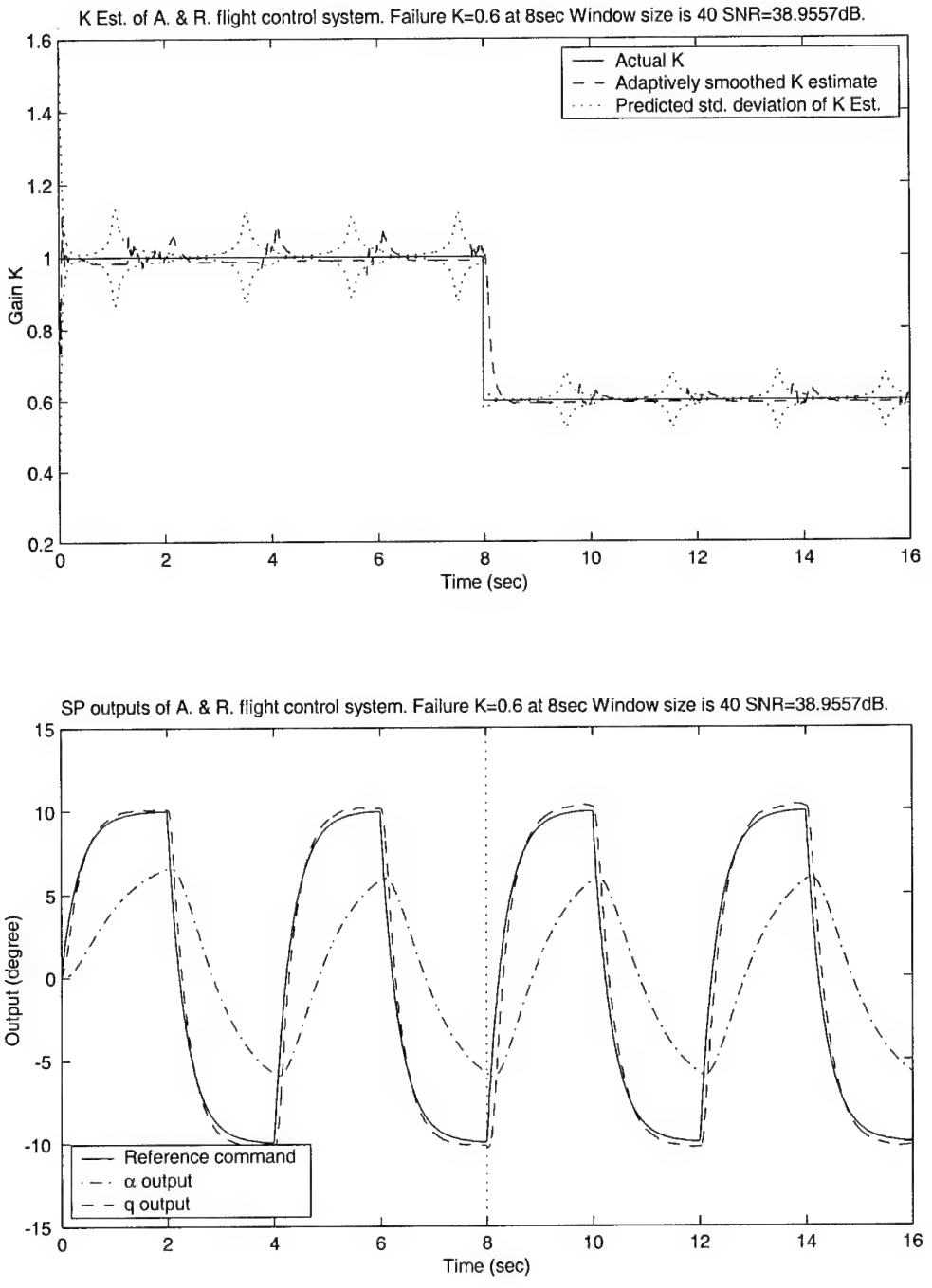


Figure 59. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. Window size is 40, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. Failure at t=8 sec. $K_1 = 0.6$.

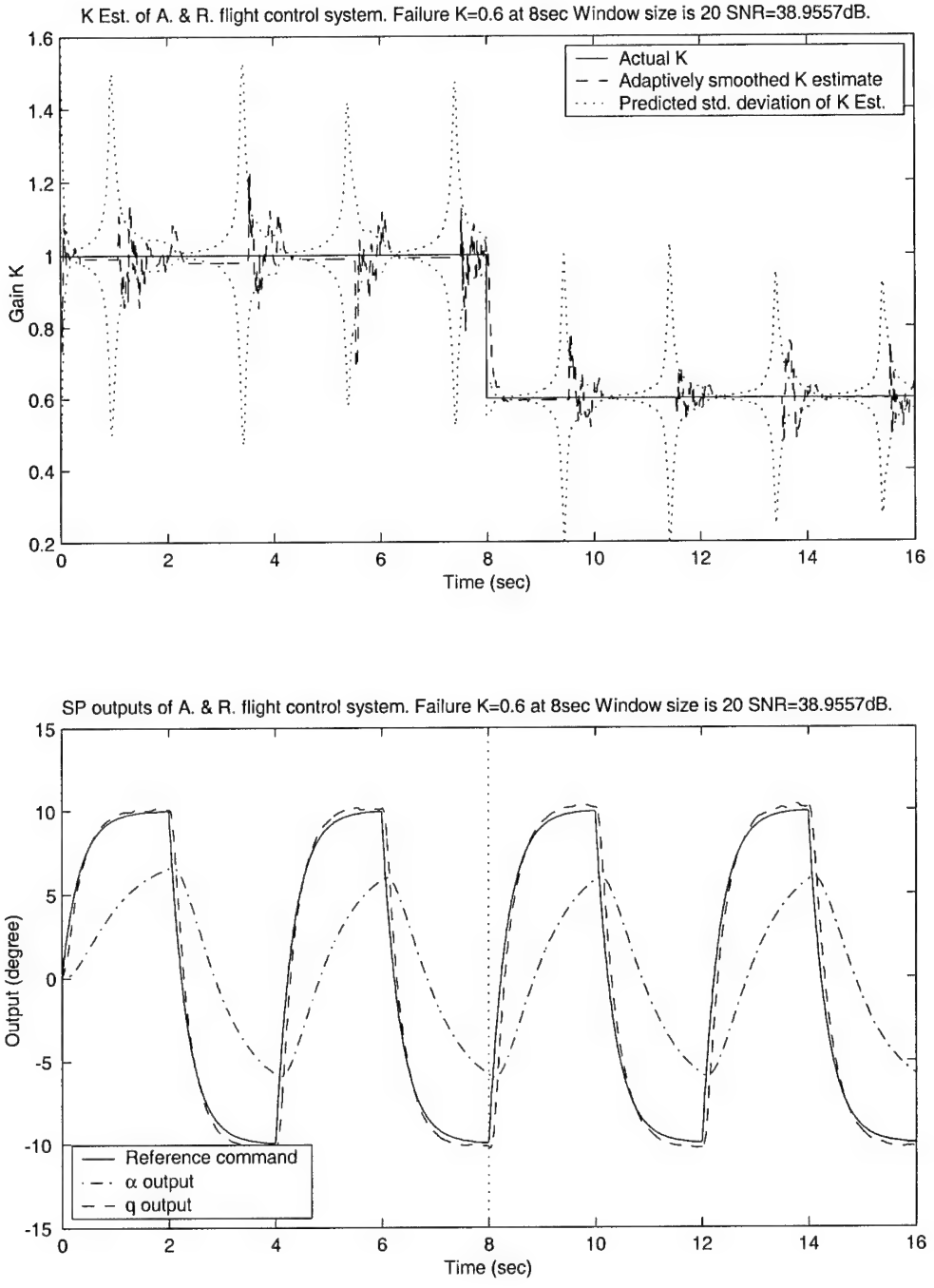


Figure 60. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. Window size is 20, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. Failure at t=8 sec. $K_1 = 0.6$.

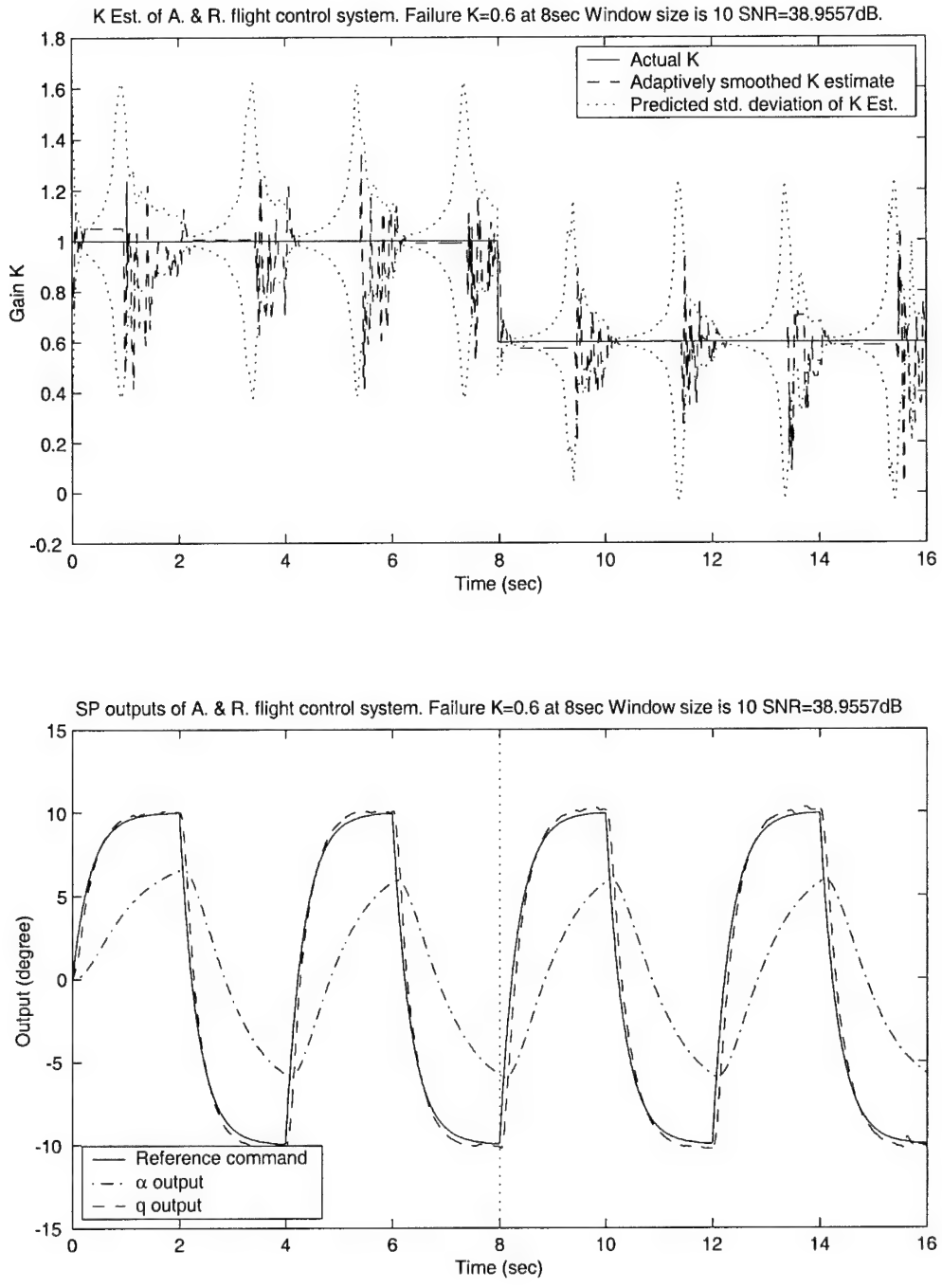


Figure 61. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm and adaptive smoother are used. Window size is 10, $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. Failure at t=8 sec. $K_1 = 0.6$.

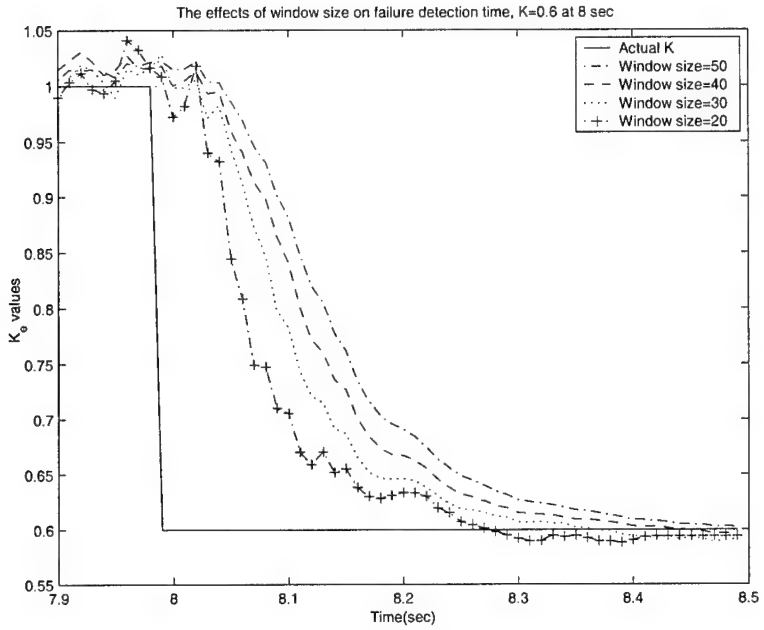


Figure 62. The effects of window size on the settling time when moving window system ID is used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

see, e.g., Eq. (90) - the new augmented dynamics are

$$\begin{aligned} \dot{x} &= Ax + bu \\ &= \begin{pmatrix} Z_\alpha & Z_q & Z_v & Z_\theta & Z_{\delta_e} \\ M_\alpha & M_q & M_v & M_\theta & M_{\delta_e} \\ X_\alpha & X_q & X_v & X_\theta & X_{\delta_e} \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{\tau} \end{pmatrix} x + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \frac{1}{\tau} \end{pmatrix} \delta_{e_c} \end{aligned} \tag{117}$$

where the state

$$x = \begin{pmatrix} \alpha \\ q \\ v \\ \theta \\ \delta_e \end{pmatrix} \tag{118}$$

The Z stability and control derivatives now are

$$Z_\alpha = -1.3433,\ Z_q = 0.9946,\ Z_v = 0,\ Z_\theta = -0.001,\ Z_{\delta_e} = -0.1525$$

the M stability and control derivatives now are

$$M_\alpha = 3.5,\ M_q = -1.0521,\ M_v = -0.0003,\ M_\theta = 0,\ M_{\delta_e} = -24.3282$$

and the X stability and control derivatives are

$$X_\alpha = 33.4778,\ X_q = -26.0592,\ X_v = -0.0119,\ X_\theta = -32.1873,\ X_{\delta_e} = 21.6603$$

The above fifth-order plant model is then the truth model used in the simulation, with the same system identification algorithm, parameter estimate smoother and PI tracking controller as used in the previous simulations. Thus, we have introduced low frequency unmodeled dynamics.

Setting the post-failure open-loop gain at $K_1 = 0.6$, window size used in the moving-window system identification algorithm is 30, and the Signal-to-Noise Ratio of the output states measurement is $SNR = 38.9557dB$. Figure 63 shows that when we include the Phugoid dynamics into our plant, the estimation performance deteriorates somewhat, however, the tracking performance does not change significantly. Thus, unmodeled low frequency dynamics are not so problematic.

7.7.2 Fourth-Order Actuator. We now exchange in the simulation the first-order actuator previously used with the fourth-order actuator specified in Eq. (91), but without the Phugoid dynamics. Thus, we have introduced high frequency unmodeled dynamics. The estimation performance and tracking performance are shown in Figure 64. The estimation performance is slightly better compared to that when the first-order actuator is used. However, tracking performance degrades significantly when the fourth-order actuator is used in the simulation experiment,

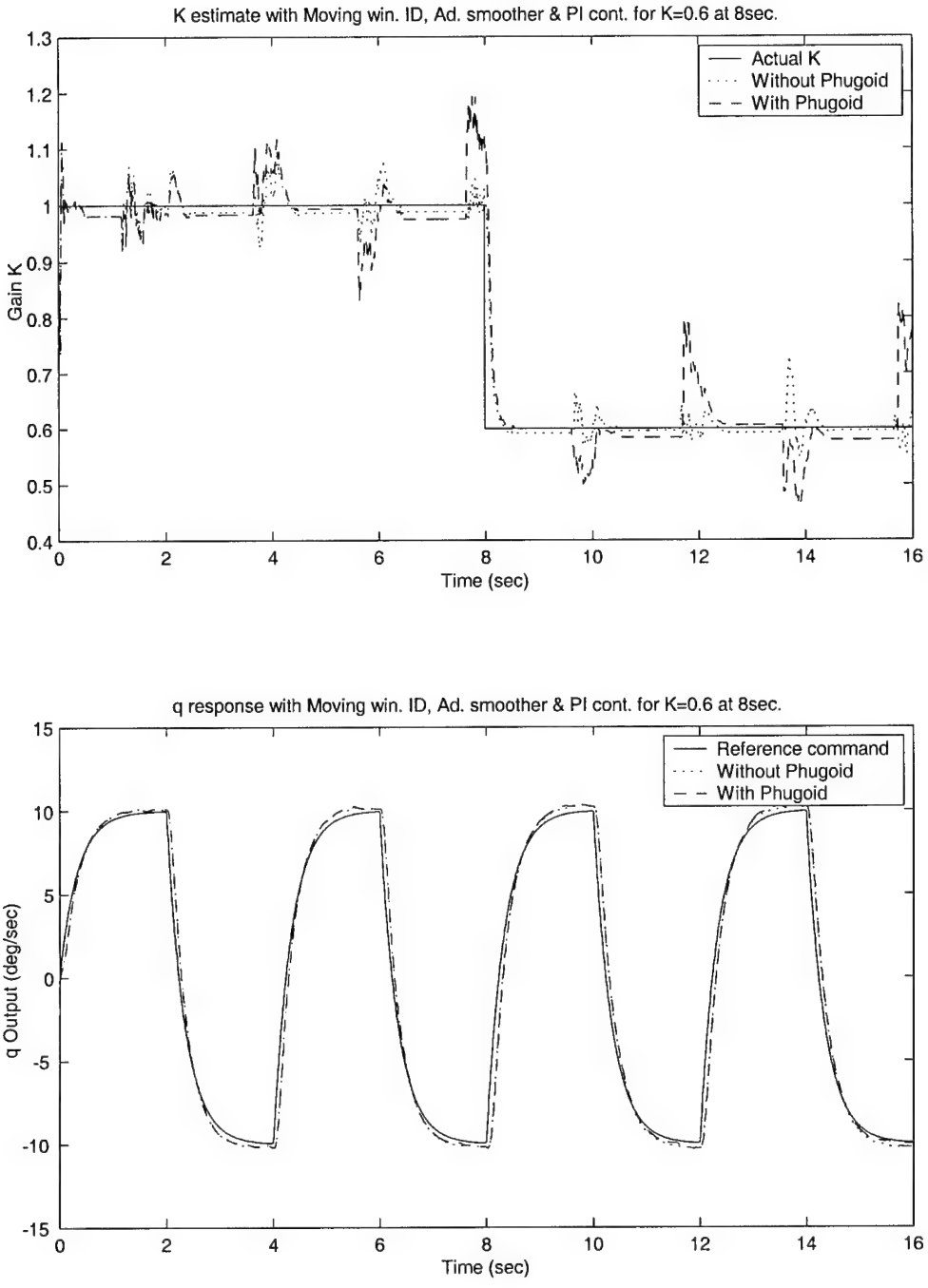


Figure 63. Phugoid dynamics are included - q response and adaptively smoothed $\widehat{K}$ when moving window system ID algorithm and adaptive smoother are used. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.

and there is "ringing" when the reference command signal passes the peak value. The latter introduces some "dither", viz., excitation, which enhances the performance of the system identification algorithm. Indeed, the system identification algorithm is not exercised in the presence of high frequency unmodeled dynamics because in the estimation algorithm we use the actual actuator deflection measurement. Thus, in Figure 64, one clearly sees the beneficial effect of dither on system identification performance.

When we change the control gain K_{δ_e} from -0.4 to -1.5 to eliminate the "ringing", the result is shown in Figure 65. The ringing is reduced, and the estimation performance is not appreciably degraded. Note, however, the lag in tracking when the fourth-order actuator is used. Thus, to account for the use of a fourth-order actuator, we need to fine tune the PI controller gains to reduce the "ringing" effect in tracking.

7.7.3 Unmodeled Phugoid and Fourth-Order Actuator Dynamics. We now include in our simulation the Phugoid dynamics and use the fourth-order actuator to assess their joint effects on identification and tracking performance. The moving-window parameter identification algorithm and the adaptive smoother are used. Figure 66 shows the estimation performance and the tracking performance when the control gain K_{δ_e} remains -0.4. Now, compare Figure 66 to Figures 63 and 64. We can see that there is still some "ringing" when the fourth-order actuator is used, but the estimation performance is better than that when only the Phugoid dynamics are included. Indeed, the inclusion of the fourth-order actuator model improves the estimation performance and mitigates the bad influence of the Phugoid.

When we change the control gain K_{δ_e} to -1.5, the results are shown in Figure 67. We can see that the "ringing" is reduced as expected, but the estimation performance is slightly degraded. However, the estimation performance is still better than

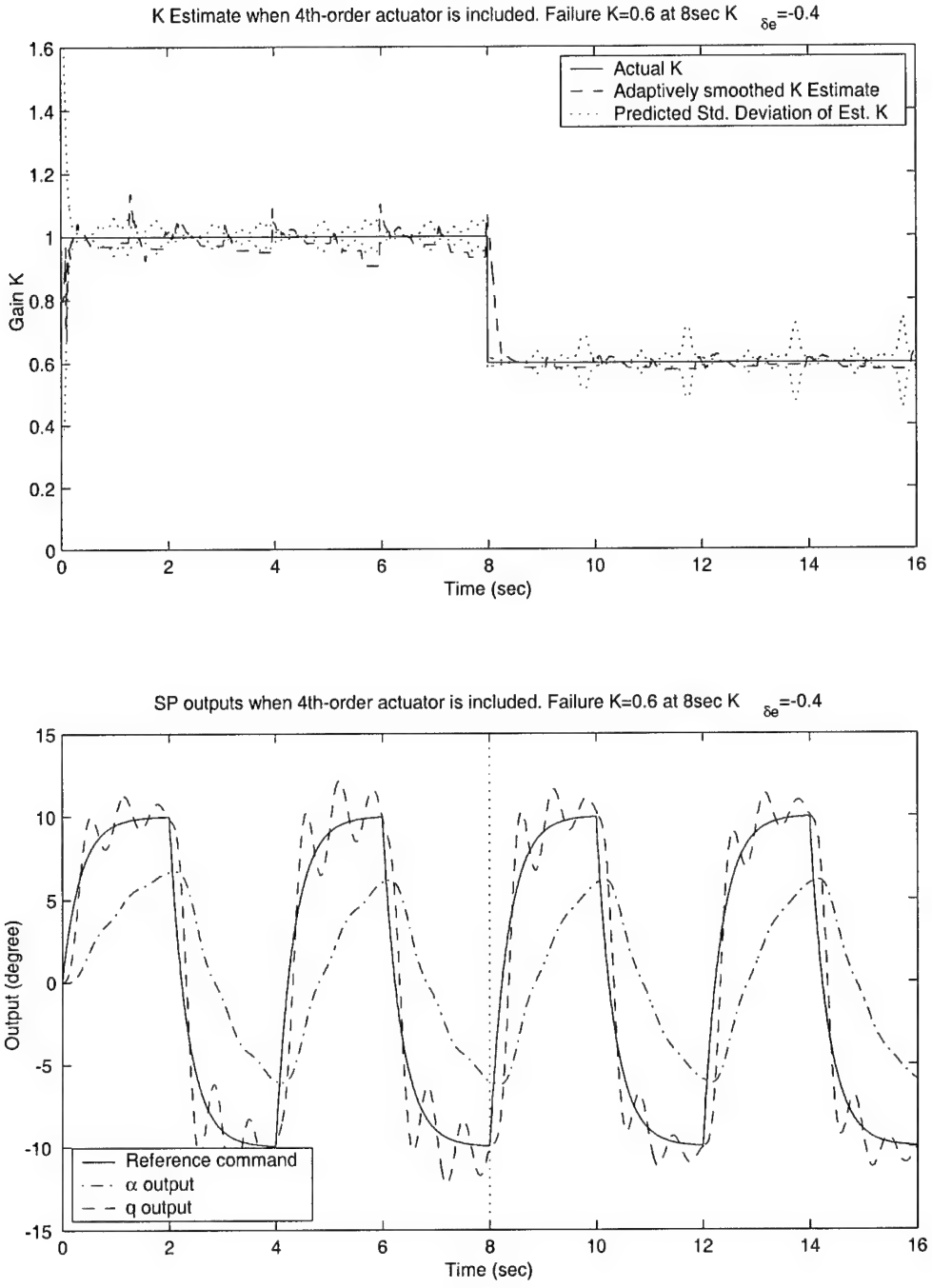


Figure 64. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm, adaptive smoother and fourth-order actuator are used. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. Failure at t=8 sec. for $K_1 = 0.6$.

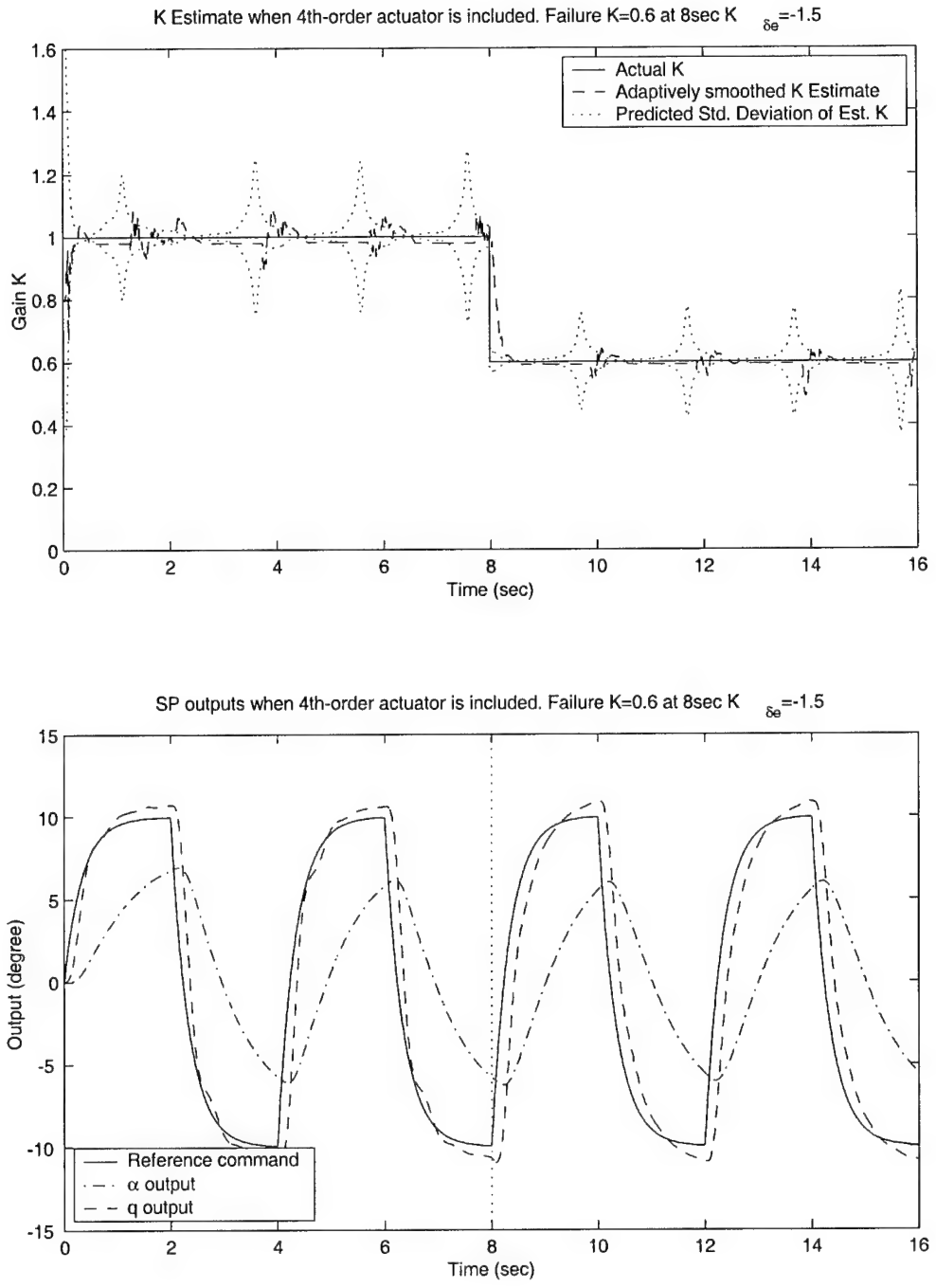


Figure 65. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm, adaptive smoother and fourth-order actuator are used. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

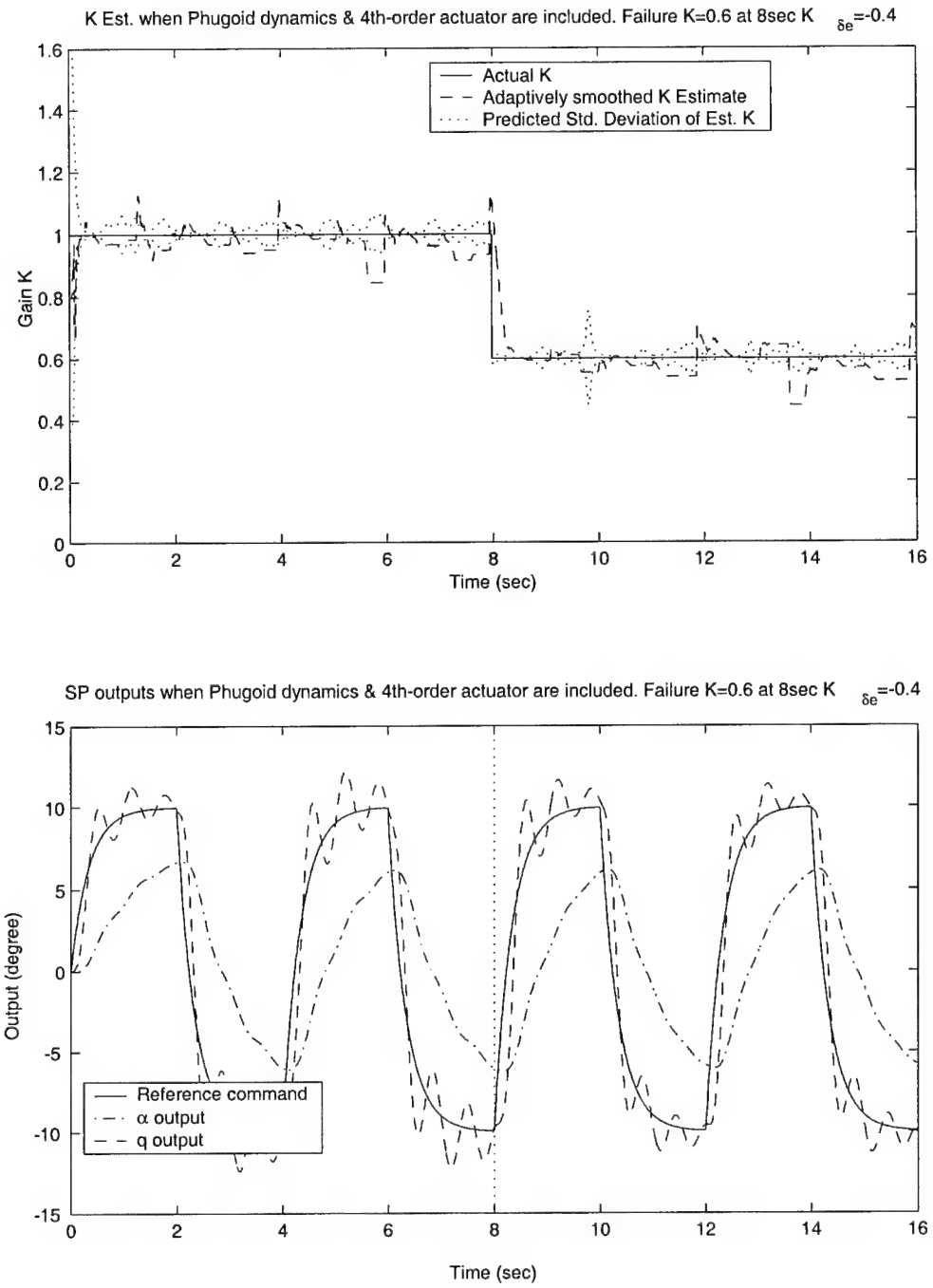


Figure 66. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm, adaptive smoother, Phugoid dynamics and fourth-order actuator are used. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

Table 7. Longitudinal Dynamics with PI Tracking Controller when modeling error Malpha=5 and Kde=-0.4.

System	**Eigenvalues** ($M_\alpha = 5$, $K_{\delta_e} = -0.4$)			
Open-loop	-20.0	-3.4325	1.0371	N/A
Closed-loop, K=1	-14.4995+15.4874j	-14.4995-15.4874j	-1.3857	-0.0107
Closed-loop, K=0.8	-14.5399+12.4700j	-14.5399-12.4700j	-1.3047	-0.0112
Closed-loop, K=0.4	-17.9041	-11.5671	-0.9100	-0.0142
Closed-loop, K=0.2	-24.3589	-5.7072	-0.2968	-0.0324
Closed-loop, K=0.15	-0.0374+0.0813j	-0.0374-0.0813j	-25.3846	-4.9360
Closed-loop, K=0.1	-26.3211	-4.3121	0.2097	0.0281

Table 8. Longitudinal Dynamics with PI Tracking Controller when modeling error Malpha=5 and Kde=-1.5.

System	**Eigenvalues** ($M_\alpha = 5$, $K_{\delta_e} = -1.5$)			
Open-loop	-20.0	-3.4325	1.0371	N/A
Closed-loop, K=1	-38.7367	-12.5252	-1.1212	-0.0123
Closed-loop, K=0.8	-41.6301	-9.7716	-0.9803	-0.0134
Closed-loop, K=0.4	-46.2408	-5.7282	-0.4012	-0.0252
Closed-loop, K=0.3	-0.0853+0.0353j	-0.0853-0.0353j	-47.2415	-4.9833
Closed-loop, K=0.25	-0.0085+0.0864j	-0.0085-0.0864j	-47.7250	-4.6534
Closed-loop, K=0.2	0.0777+0.0185j	-0.0777-0.0185j	-48.1982	-4.3526

that when only the Phugoid dynamics are included, but now there is a slight lag in tracking.

7.7.4 Parameter Modeling Error. We now introduce modeling "error" in the M_α stability derivative. Thus, in the simulation we set $M_\alpha = 5$ after the failure. Table 7 and 8 show the poles of the open-loop plant and the poles of the closed-loop system when the fixed PI controller is used with $M_\alpha = 5$. In the presence of modeling error ($M_\alpha = 5$), as the loop gain K is lowered from a value of 1 to a value of $K_1 \approx 0.1$ (when $K_{\delta_e} = -0.4$) and to a value of $K_1 \approx 0.2$ (when $K_{\delta_e} = -1.5$), the closed-loop system reverts to instability. As we recall from Section 6.2, without parametric modeling error, the fixed PI controller can stabilize the open-loop plant until $K_1 = 0.08$. The inclusion of parameter modeling error degrades the stability robustness of the flight control system.

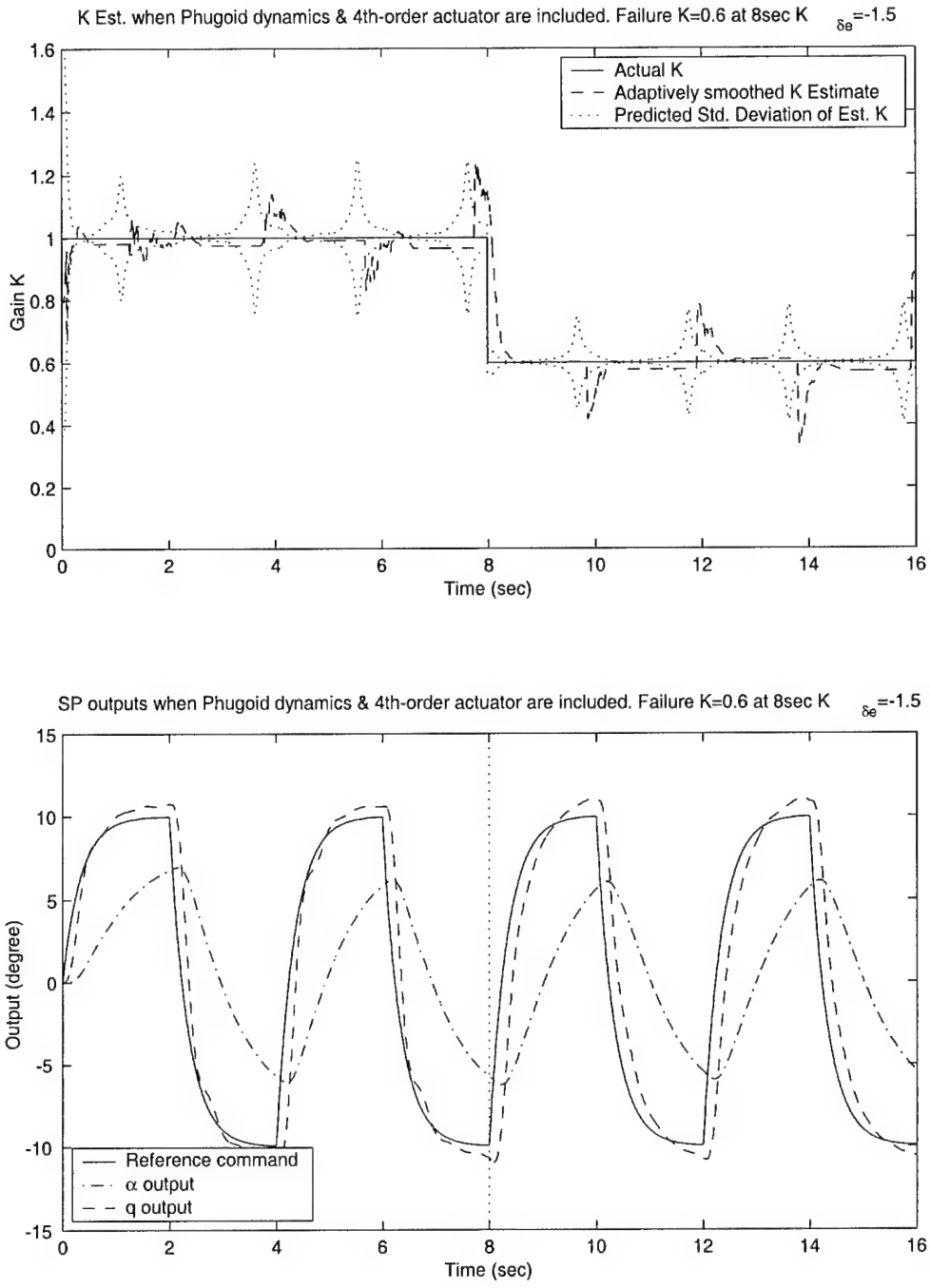


Figure 67. Smoothed $\widehat{K}$ and tracking performance when moving window system ID algorithm, adaptive smoother, Phugoid dynamics and fourth-order actuator are used. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

Table 9. Estimation performance without/with Modeling Error.

Window Length	Without Modeling Error				With Modeling Error after Failure			
	Before Failure		After Failure		Before Failure		After Failure	
	$M_\alpha = 3.5$		$M_\alpha = 3.5$		$M_\alpha = 3.5$		$M_\alpha = 5$	
	%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$
40	47.44%	1.0277	36.68%	2.2823	47.44%	1.0261	1.50%	16.2354
30	55.90%	1.0345	44.50%	1.4514	55.90%	1.0336	2.88%	10.2572
20	63.25%	0.8615	57.13%	1.1677	63.25%	0.8610	4.13%	5.9566
10	65.11%	0.8179	60.25%	0.9802	65.11%	0.8176	11.63%	2.7440
8	63.56%	0.9170	59.88%	0.9990	63.68%	0.9166	16.13%	2.7000
5	58.79%	1.3711	57.63%	1.1323	58.79%	1.3718	28.25%	1.7006

Table 9 show the performance of the system identification algorithm after the failure in the presence of parametric modeling error. The percentage of the parameter estimates $\widehat{K}$ falling inside the predicted 1σ bounds, and the ratio $\sigma_{K_e}/\overline{\sigma}_K$, are shown for window sizes w_l of $40, 30, 20, 10, 8, 5$. Here σ_{K_e} is the experimentally obtained variance of the parameter estimation error, and $\overline{\sigma}_K$ is the average predicted variance of the parameter estimation error:

$$\sigma_{K_e} = \sqrt{\frac{1}{N_s}\sum_{k=1}^{N_s}(K_k - \widehat{K}_k)^2}$$

and

$$\overline{\sigma}_K = \sqrt{\frac{1}{N_s}\sum_{k=1}^{N_s}\sigma_{K_k}^2}$$

and where N_s is the number of samples in our experiments. Thus,

$$N_s = \begin{cases} 1600 - w_l & \text{for } 0 < t \leq 16 \text{ second}, \quad M_\alpha = 3.5, \quad \text{without modeling error} \\ 800 - w_l & \text{for } 0 < t \leq 8 \text{ second}, \quad M_\alpha = 3.5, \quad \text{with modeling error} \\ 800 & \text{for } 8 < t \leq 16 \text{ second}, \quad M_\alpha = 5, \quad \text{with modeling error} \end{cases}$$

where w_l is the moving-window length used in the system identification algorithm.

Figures 68 to 71 show the estimation and tracking performance when the moving-window system identification algorithm and no smoothing filter is used, without modeling error and with modeling error after the control surface failure. The

change in M_α due to modeling error significantly degrades the estimation performance. This is reflected in an increase in the parameter estimation error variance and in the ratio $\sigma_{K_e}/\overline{\sigma}_K \neq 1$. Moreover, from Table 9, we can see that in the presence of parametric modeling error, the shorter the windows length is, the higher the percentage of K estimates falling inside the predicted 1σ bound and the closer the predicted parameter estimation error variance is to the experimentally obtained parameter estimation error variance. In addition, the parameter estimation error variance increases, which indicates that the parameter estimate needs to be heavily smoothed before it is sent to the controller. Therefore, considering the estimation and tracking performance trade off, the "optimal" choice of window size is 30.

7.7.5 Unmodeled Phugoid Dynamics, Fourth-Order Actuator, and Parameter Modeling Error. We now include in our realistic simulation the Phugoid dynamics, use the fourth-order actuator model, and allow for a post failure parameter modeling error ($M_\alpha = 5$). We investigate their cumulative effect on identification and tracking performance. The moving-window parameter identification algorithm and the adaptive smoother are used. The window size is 30, as discussed in the pervious section. Figures 72 and 73 show the estimation performance and the tracking performance without and with the adaptive smoother, respectively, with the original control gain $K_{\delta_e} = -0.4$. Without the adaptive smoother, the effect of modeling error on the estimation and tracking performance is very pronounced. Thus, the adaptive smoother performs a crucial function in the presence of modeling error.

When we change the control gain K_{δ_e} to -1.5, the results are shown in Figures 74 and 75 with and without the adaptive smoother, respectively. The "ringing" is now reduced, but, due to the change in the control gain K_{δ_e}, the estimation performance is degraded. Without the adaptive smoother, the effect of modeling error on the estimation and tracking performance is more severe than with the adaptive smoother in place. Comparing Figures 75 and 67, we clearly see that the parametric modeling error degraded both the estimation and the tracking performance.

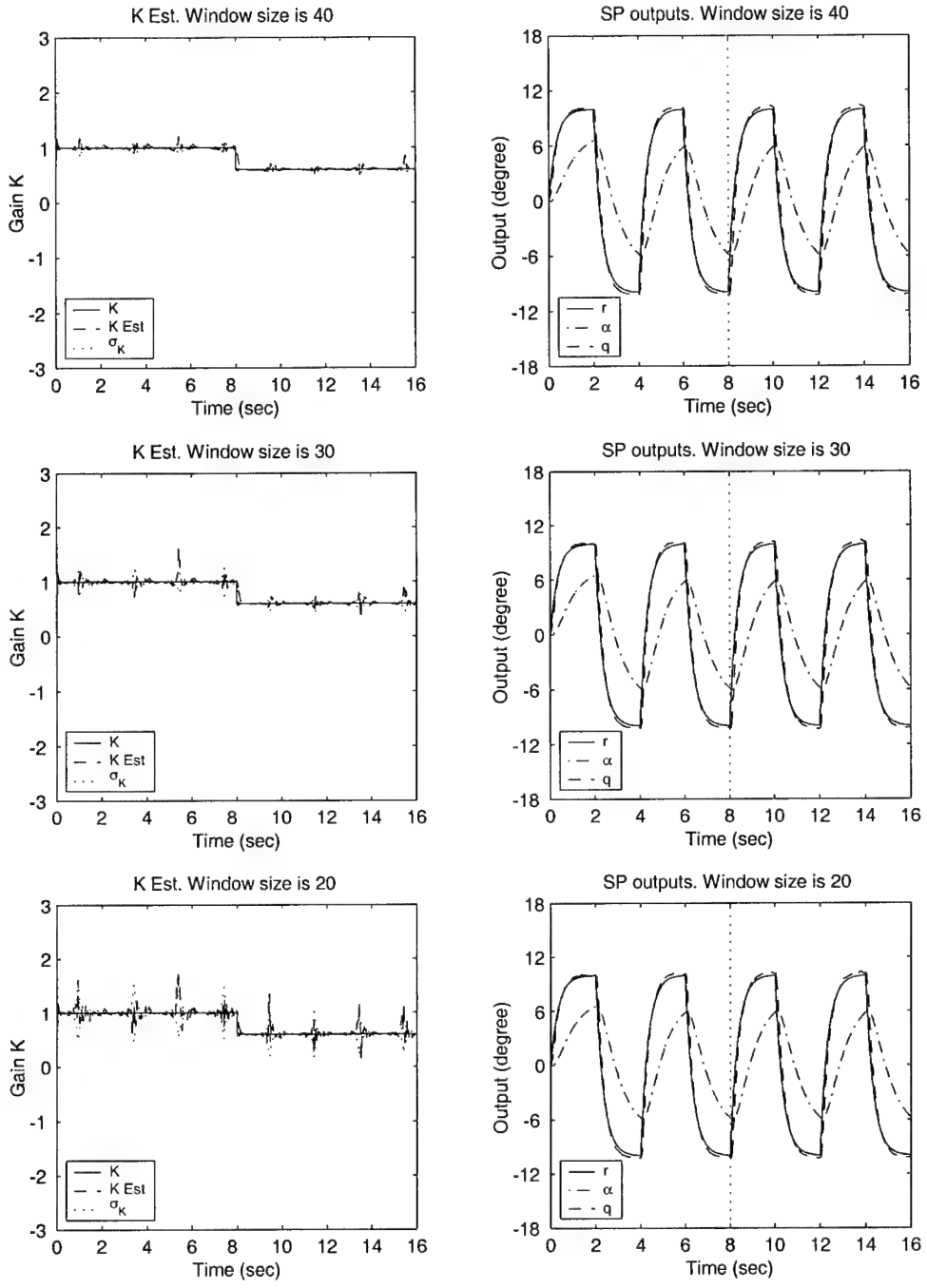


Figure 68. Parameter estimate $\widehat{K}$ and tracking performance when the moving window system ID algorithm is used without smoother. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$. Window lengths are 40, 30 and 20. No modeling error.

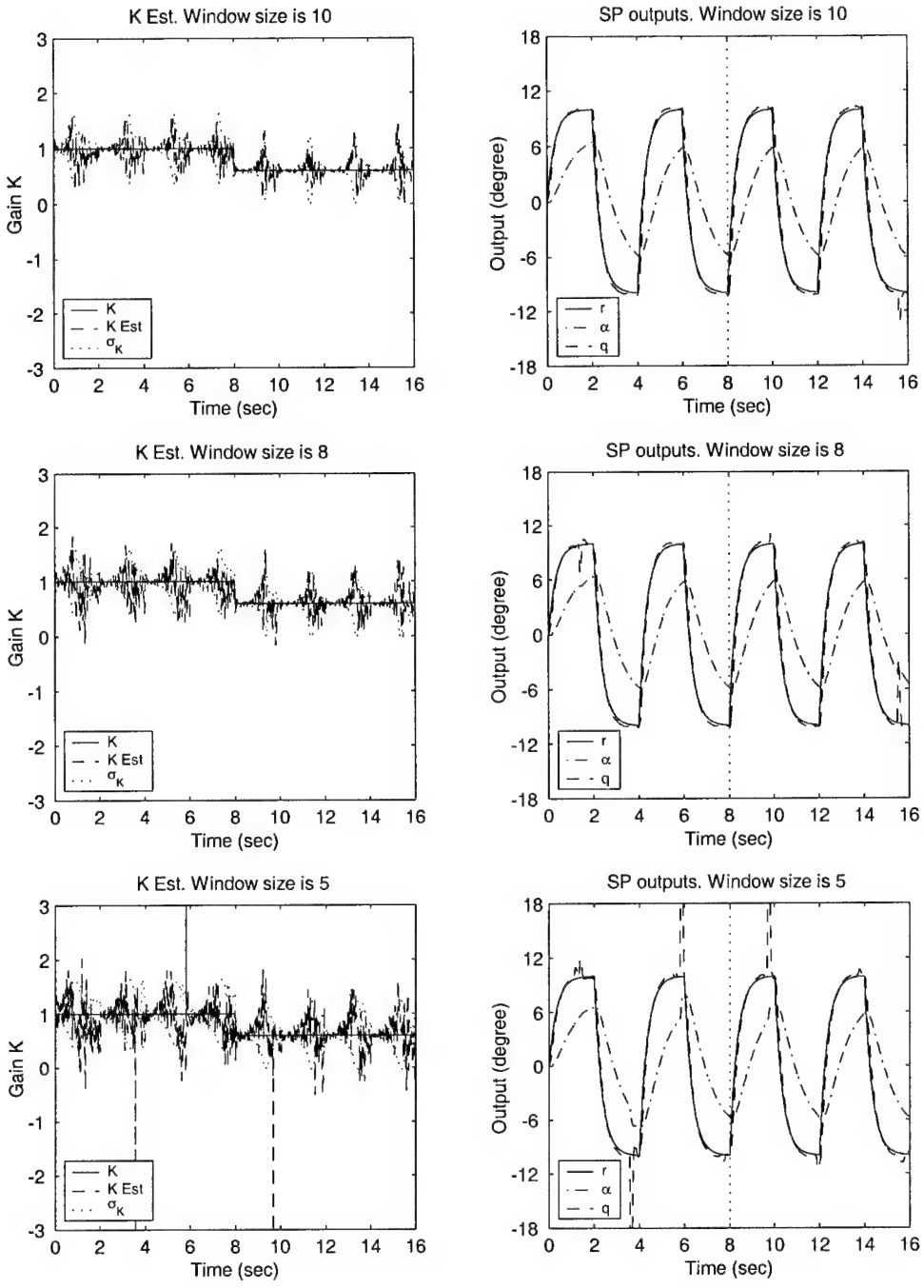


Figure 69. Parameter estimate $\widehat{K}$ and tracking performance when the moving window system ID algorithm is used without smoother. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$. Window lengths are 10, 8 and 5. No modeling error.

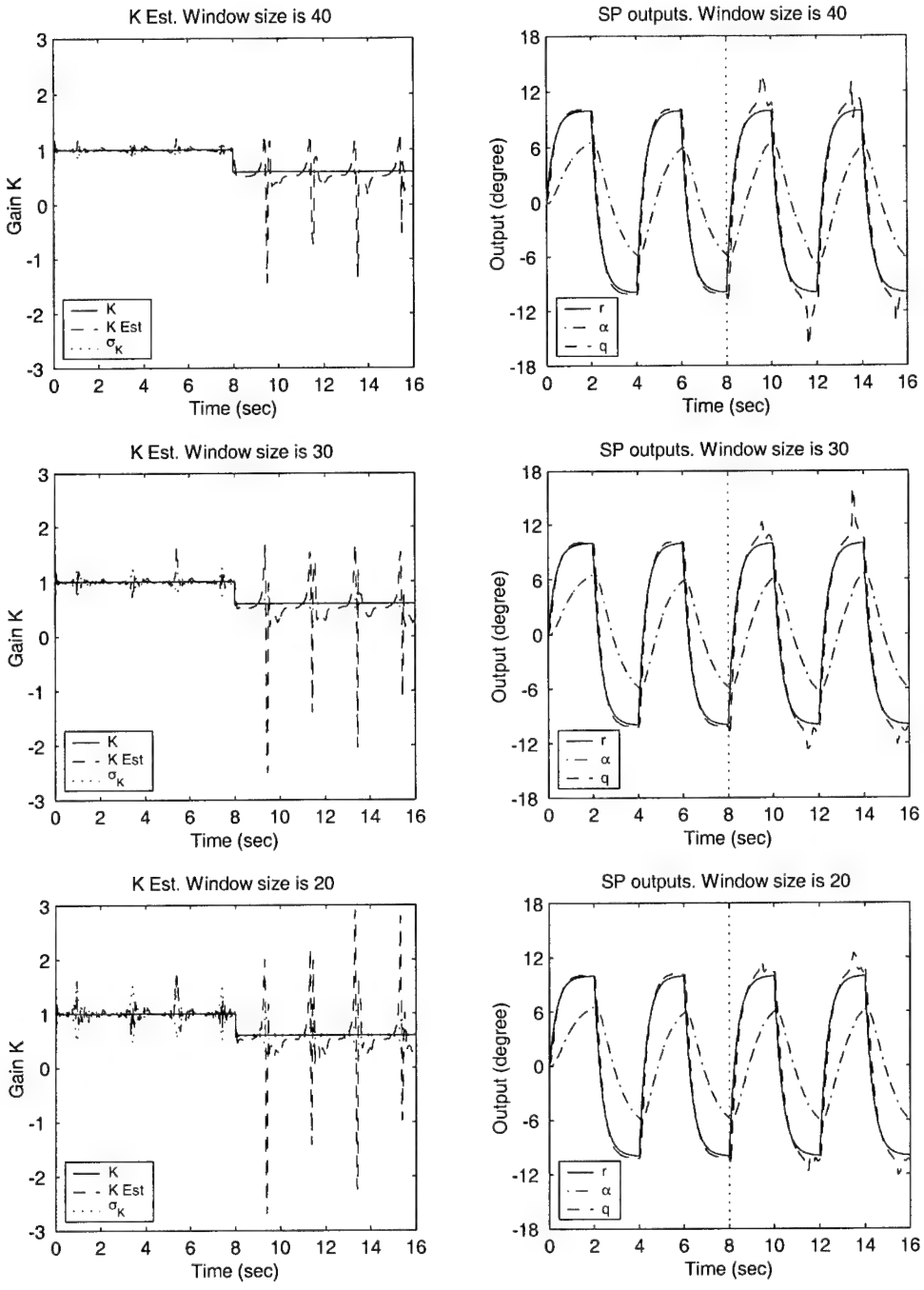


Figure 70. Parameter estimate $\widehat{K}$ and tracking performance when the moving window system ID algorithm is used without smoother. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$. Window lengths are 40, 30 and 20. With modeling error ($M_\alpha = 5$) after failure.

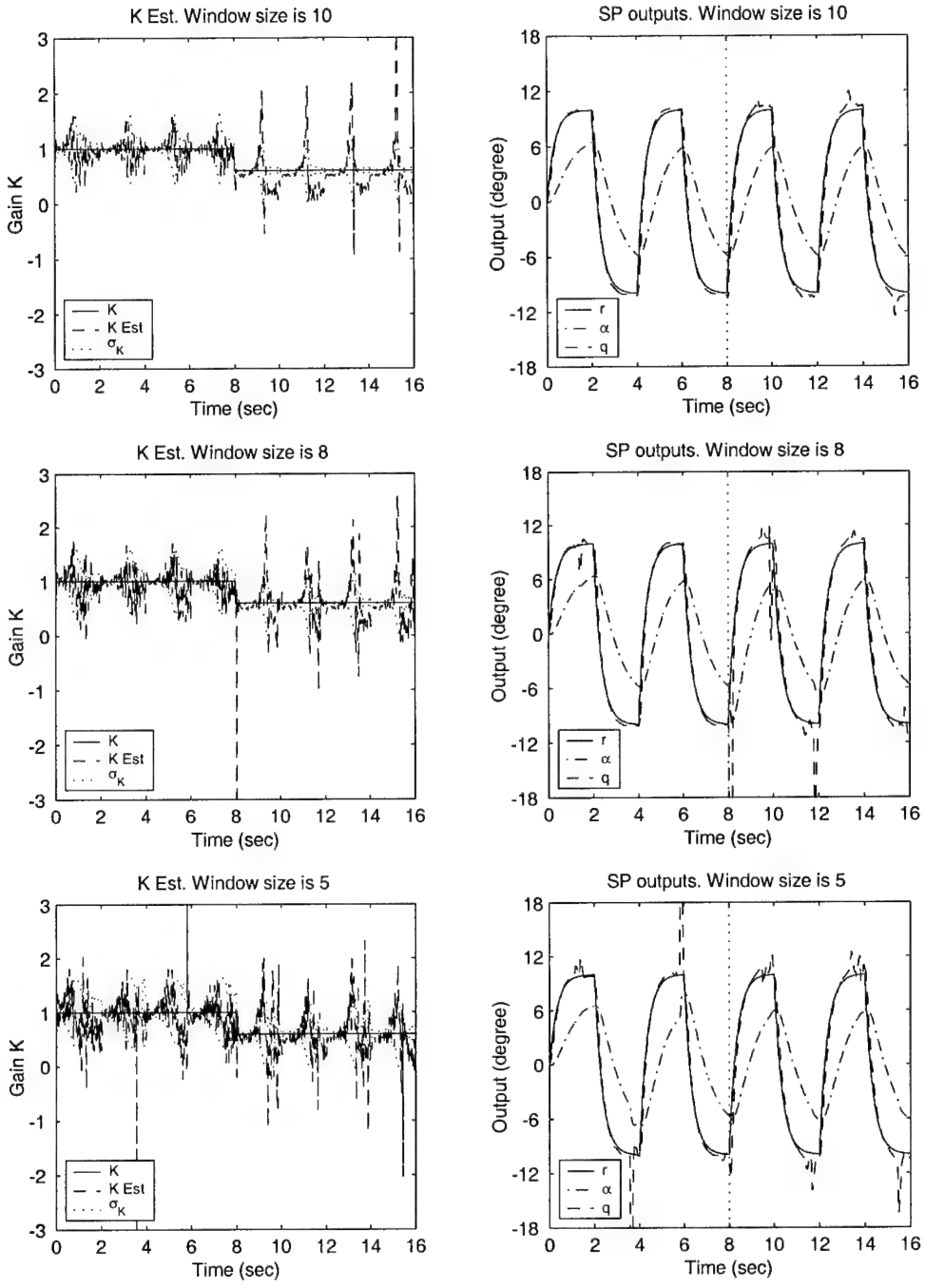


Figure 71. Parameter estimate $\widehat{K}$ and tracking performance when the moving window system ID algorithm is used without smoother. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$. Window lengths are 10, 8 and 5. With modeling error ($M_\alpha = 5$) after failure.

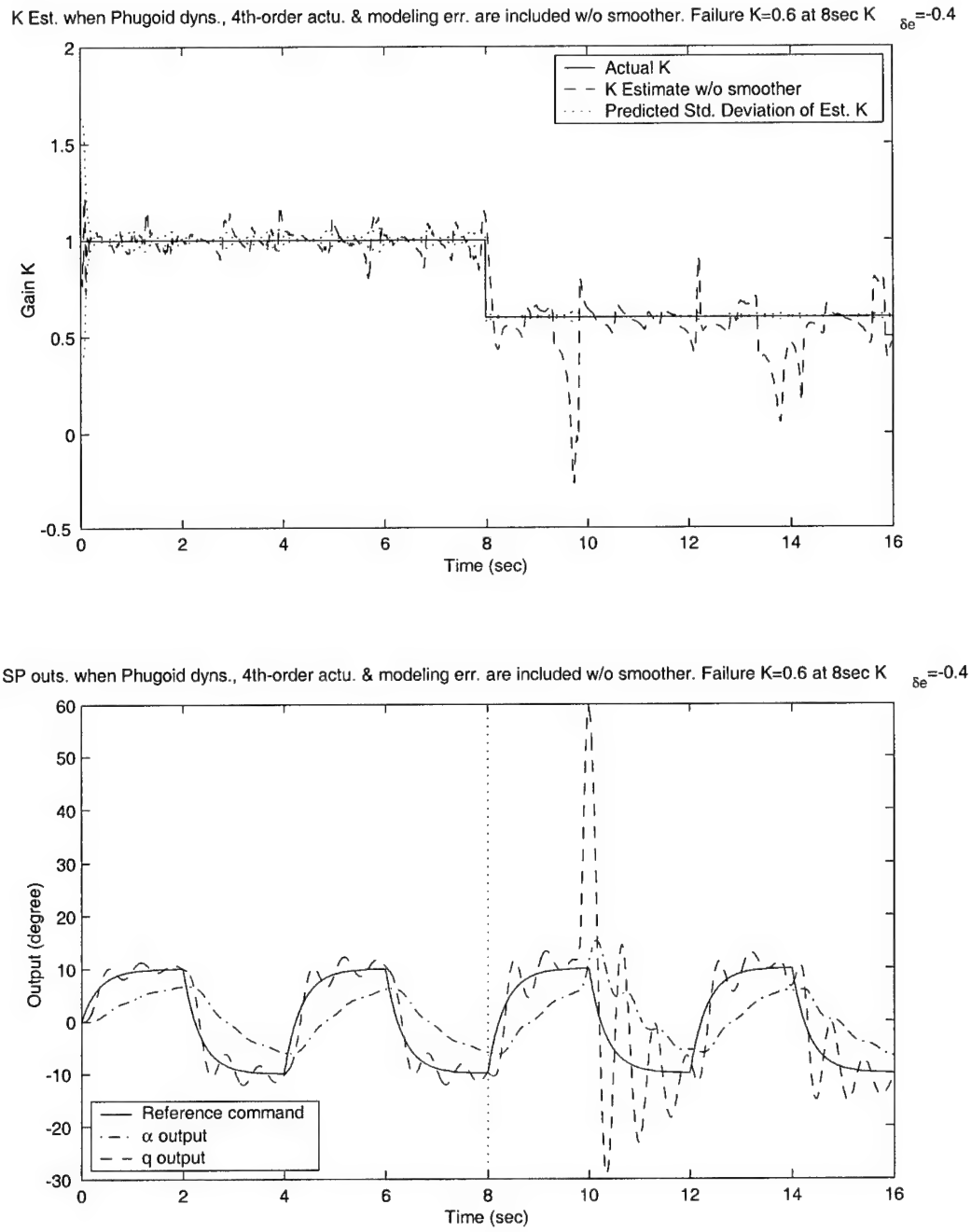


Figure 72. Parameter estimate $\widehat{K}$ and tracking performance. Moving window system ID algorithm ($wl = 30$), without smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.

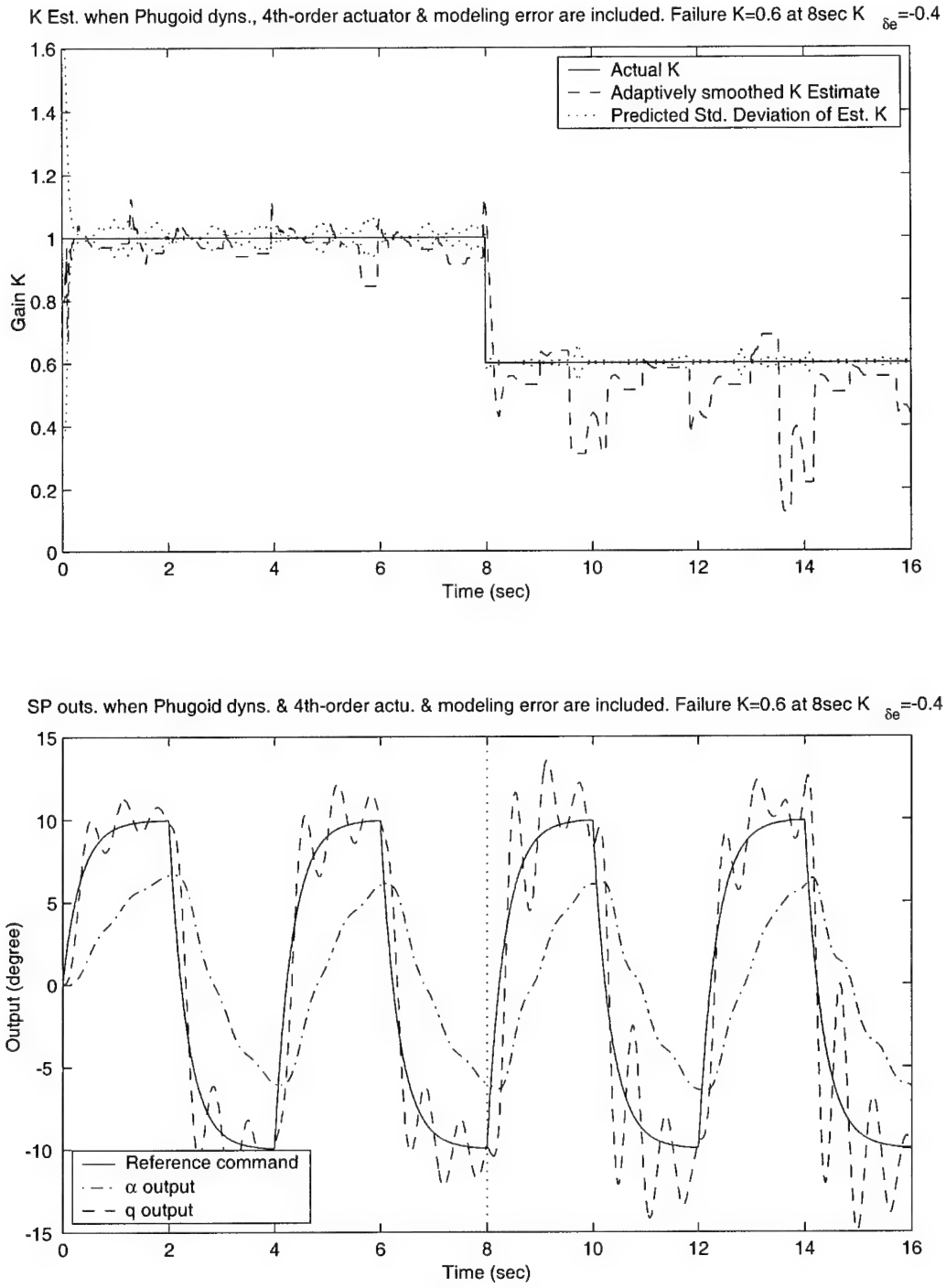


Figure 73. Smoothed $\widehat{K}$ and tracking performance. Moving window system ID algorithm ($wl = 30$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

Figures 77 to 79 show the tracking performance of the fixed PI controller, and the three-module adaptive and reconfigurable controller (moving-window identification algorithm with adaptive smoother), when the Phugoid dynamics, the fourth-order actuator model, and the post failure modeling error ($M_\alpha = 5$) are included. We also set the control gain to the original $K_{\delta_e} = -0.4$, and the control surface failure index is $K_1 = 0.6$, $K_1 = 0.4$, $K_1 = 0.2$. Before the failure, the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller are similar. However, after the failure, the adaptive and reconfigurable controller outperforms the fixed PI controller, in particular, in the severe failure case of $K_1 = 0.2$, when the fixed PI controller causes a departure.

Finally, Figure 76 shows the elevator deflection and deflection rate when the fixed PI controller, and the three-module adaptive and reconfigurable controller are used. Phugoid dynamics, the fourth-order actuator model, and the post failure parameter modeling error ($M_\alpha = 5$) are included. The case when there is no parameter modeling error is also shown ($M_\alpha = 3.5$). The control gain is set to $K_{\delta_e} = -1.5$. Before the failure, the elevator deflection and deflection rate from the fixed PI controller and the adaptive and reconfigurable controller are similar. After the failure and in the presence of parameter modeling error, the elevator deflection is reasonable ($-6.46° < \delta_e < 6.32°$), however the elevator deflection rate after the failure and when adaptive and reconfigurable control is used is ± 65 deg/sec at $t \approx 10$ sec. This is due to the fact that the smoothed K estimate suddenly drops to $\widehat{K}_{smooth} \approx 0$ - see, e.g., Figure 75. When there is no parameter modeling error after the failure, the elevator deflection rate when the adaptive and reconfigurable controller is used is in the range of ± 24 deg/sec.

When we change the control gain K_{δ_e} to -1.5, the results are shown in Figures 80 to 82. The adaptive and reconfigurable controller outperforms the fixed PI controller in all failure cases. In the case of a severe failure ($K_1 = 0.2$), the adaptive

and reconfigurable controller shows some lag in tracking, but the fixed PI controller causes a departure.

7.7.6 Unmodeled Phugoid Dynamics, Fourth-Order Actuator, and Parameter Modeling Error: Window Length Effect. We include the unmodeled Phugoid dynamics, fourth-order actuator, and post failure parameter modeling error (M_α=5) into our three-module adaptive and reconfigurable flight control system simulation. The performance of the system identification algorithm is assessed when we set the window size w_l used in the moving-window system identification algorithm at 120, 100, 80, 60, 50, 40 and 30. The post-failure open-loop gain is set at $K_1 = 0.6$, 0.4, and 0.2, and the Signal-to-Noise Ratio of the measurement is $SNR = 38.9557dB$. Figures 83 to 89 show the estimation and tracking performance in each case. A larger window size yields better parameter estimates, but the tracking performance is reduced. Indeed, using longer windows reduces the fluctuations in the parameter estimate, and bursting. However, longer windows bring about a delay in the estimation of the loop gain after the failure. From Table 10, we can see that in the presence of parametric modeling error, the shorter the windows size is, the higher the percentage of K estimates falling inside the predicted 1σ bound. At the same time, the parameter estimation error variance increases, which indicates that the parameter estimate needs to be heavily smoothed before it is sent to the controller. Therefore, when the sampling rate is 100 Hz, and considering the estimation and tracking performance trade off, the "optimal" choice of window size for inner loop flight control is $w_l = 30$. For reference purpose, Table 11 is reproduced for the case when there are no modeling error. The adverse effect on estimation performance of modeling error is clearly visible.

7.7.7 Discussion. From the simulation experiments, we conclude that the unmodeled dynamics adversely affect the system identification algorithm. The Phugoid dynamics will degrade the estimation performance, but preserve the track-

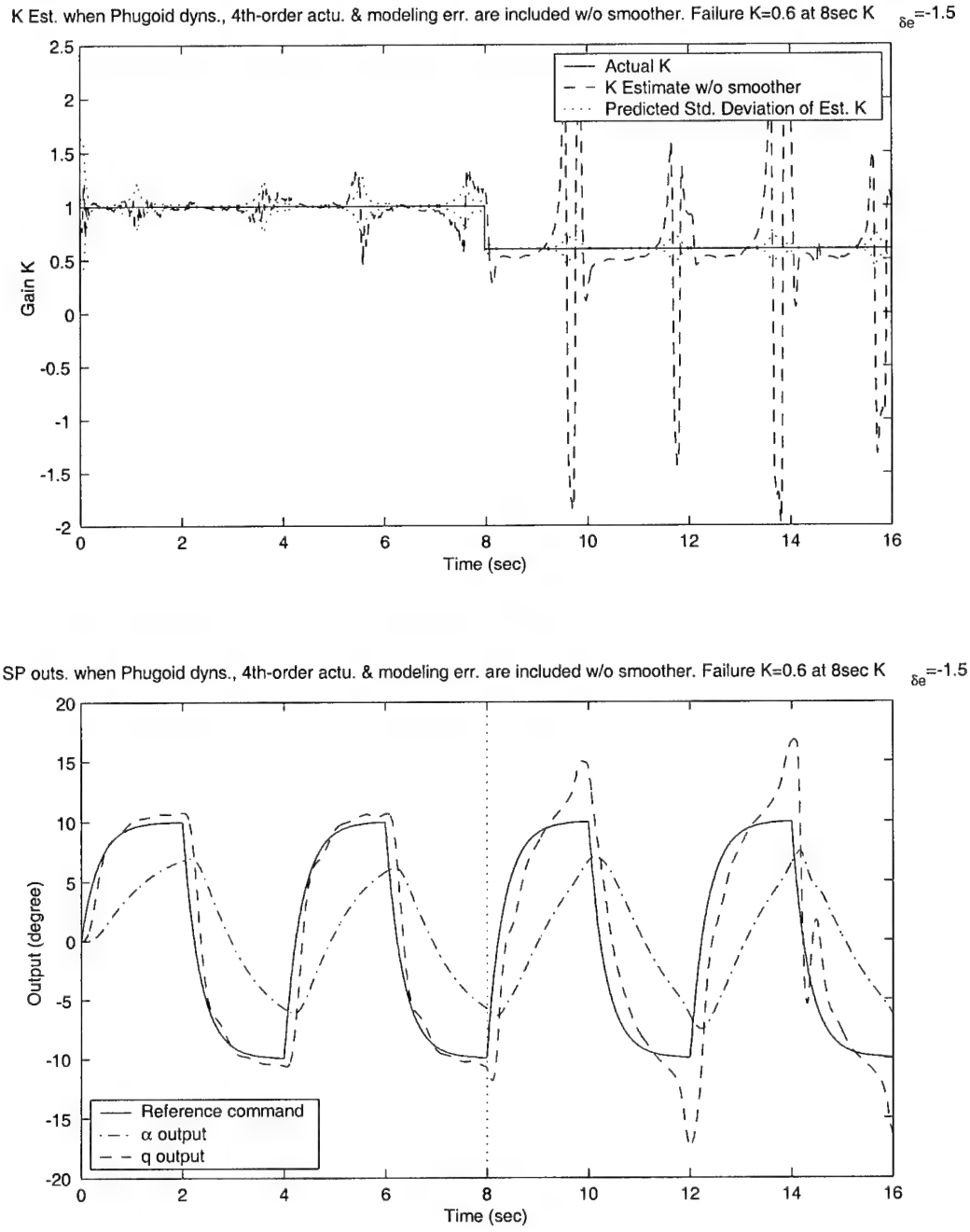
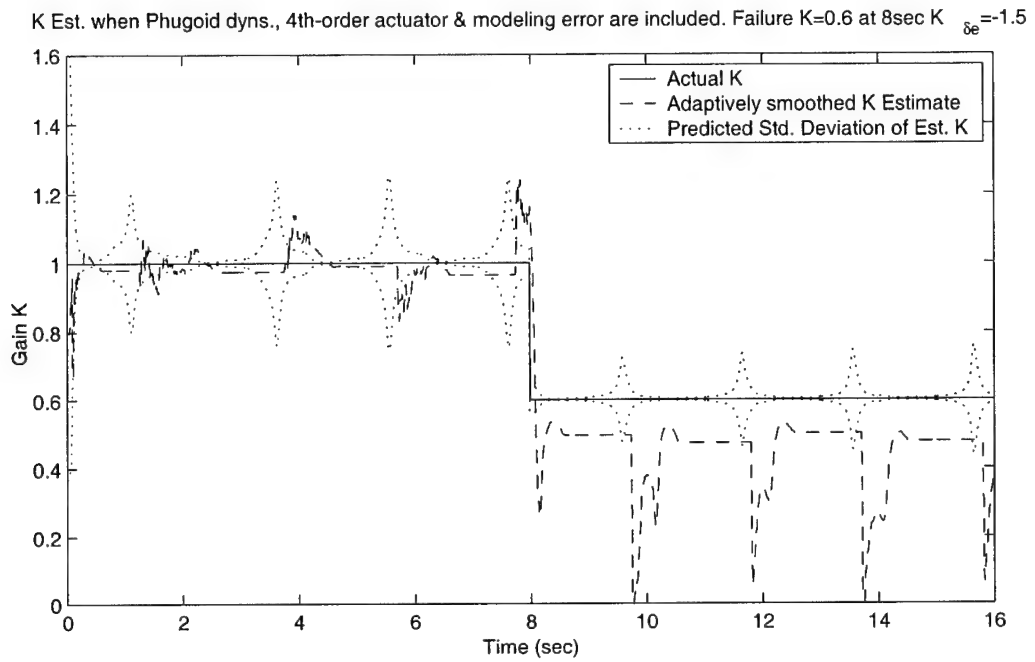


Figure 74. Parameter estimate $\widehat{K}$ and tracking performance. Moving window system ID algorithm ($wl = 30$), without smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.



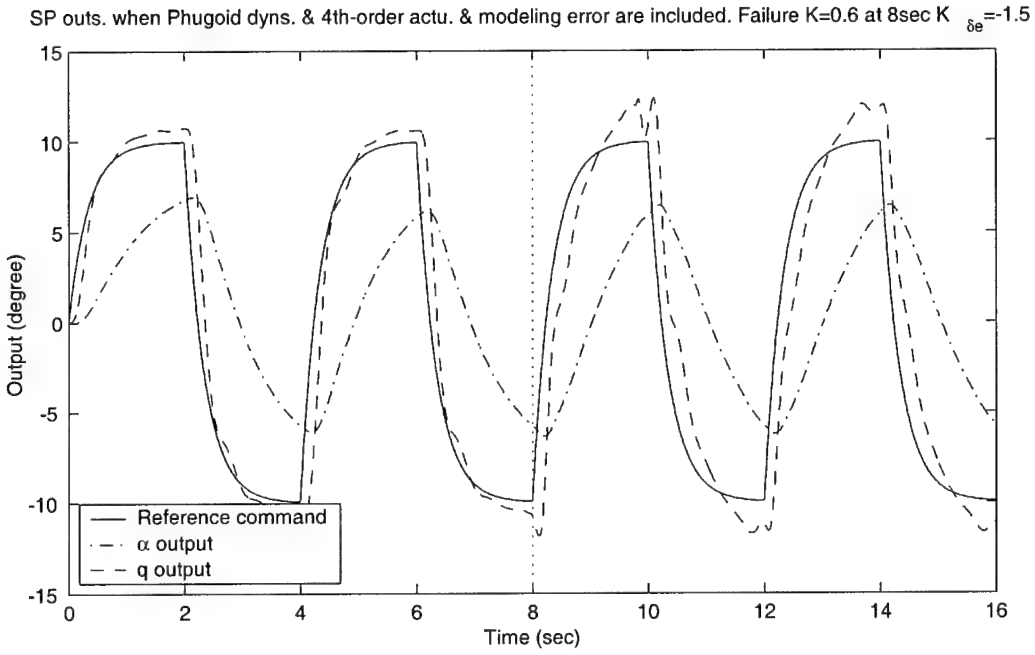


Figure 75. Smoothed $\widehat{K}$ and tracking performance. Moving window system ID algorithm ($wl = 30$), with adaptive sommther. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}\,/\,\text{sec}$. $K_1 = 0.6$.

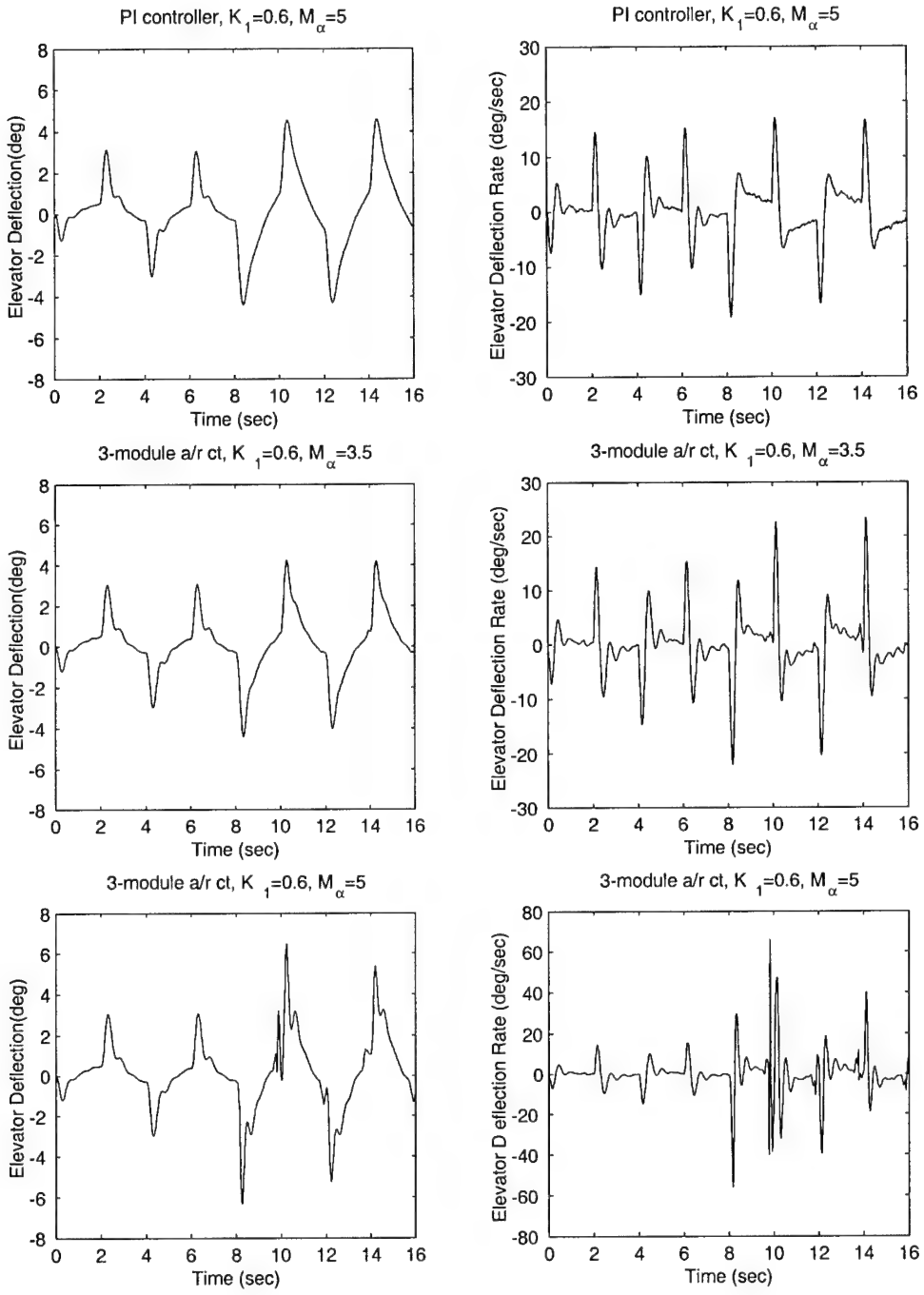


Figure 76. Comparison of the elevator deflection and deflection rate when the fixed PI controller and the adaptive and reconfigurable controller are used. Phugoid dynamics, fourth-order actuator model and with and without parametric modeling error ($M_\alpha = 5$ after failure) included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$.

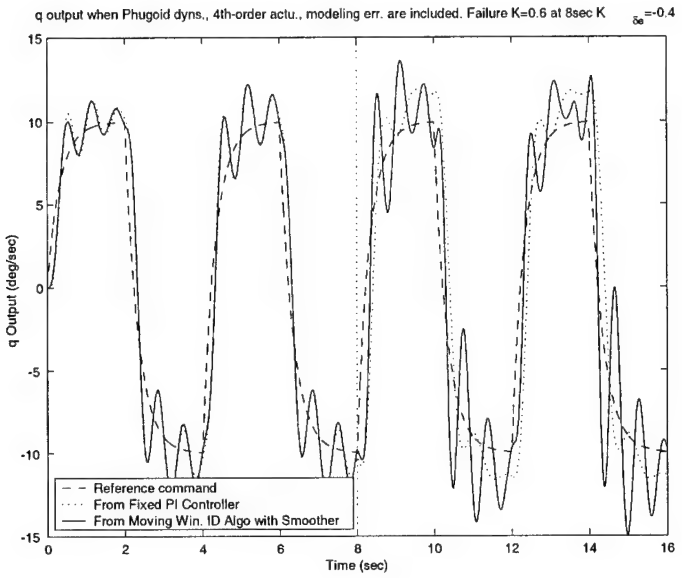


Figure 77. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

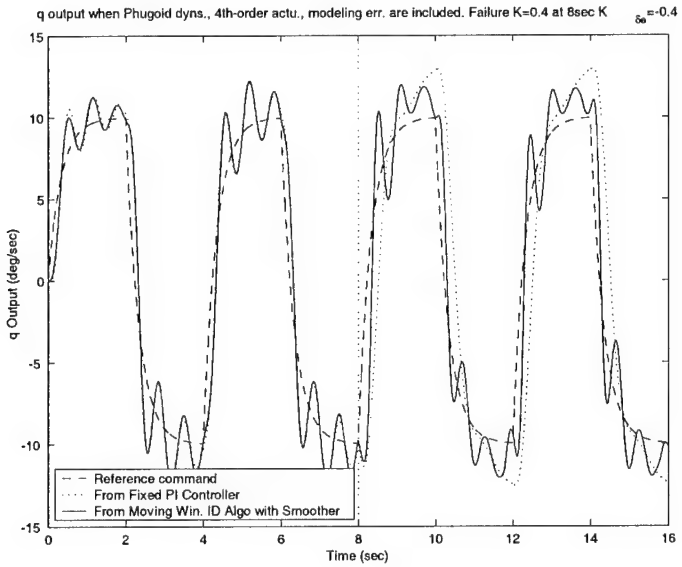


Figure 78. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.4$.

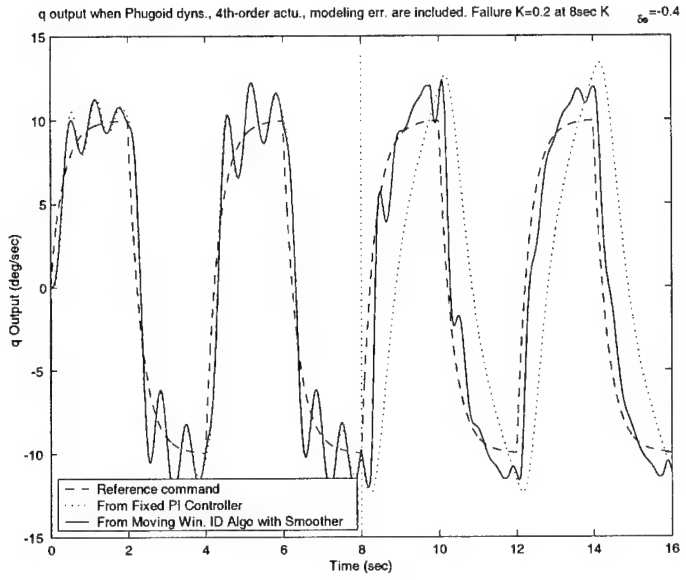


Figure 79. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -0.4$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.2$.

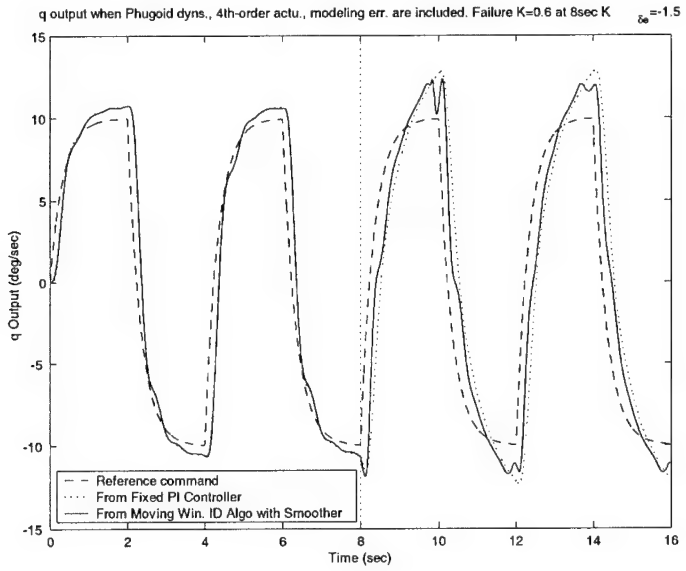


Figure 80. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$.

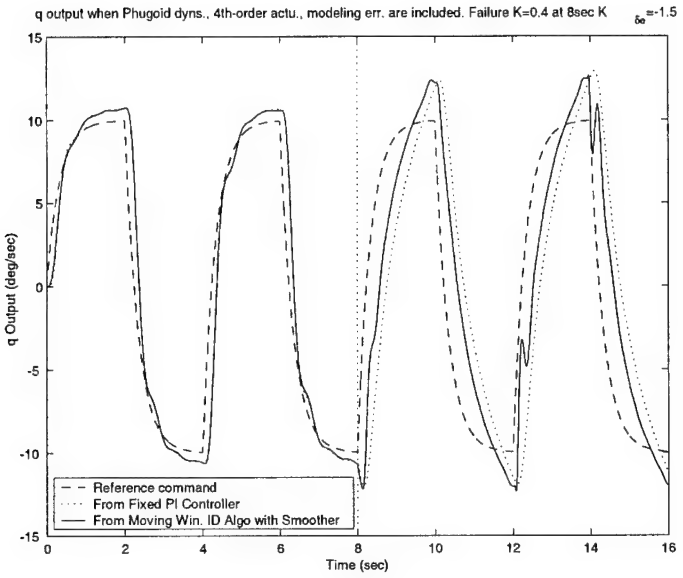


Figure 81. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.4$.

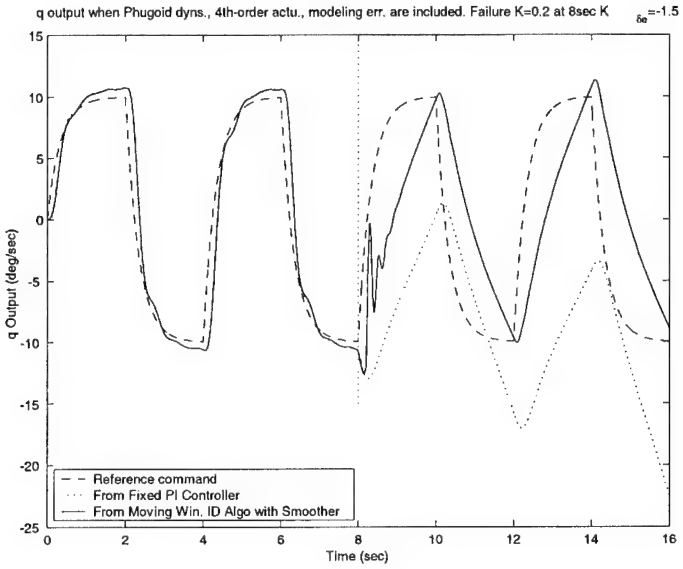


Figure 82. Comparison of the tracking performance of the fixed PI controller and the adaptive and reconfigurable controller. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.2$.

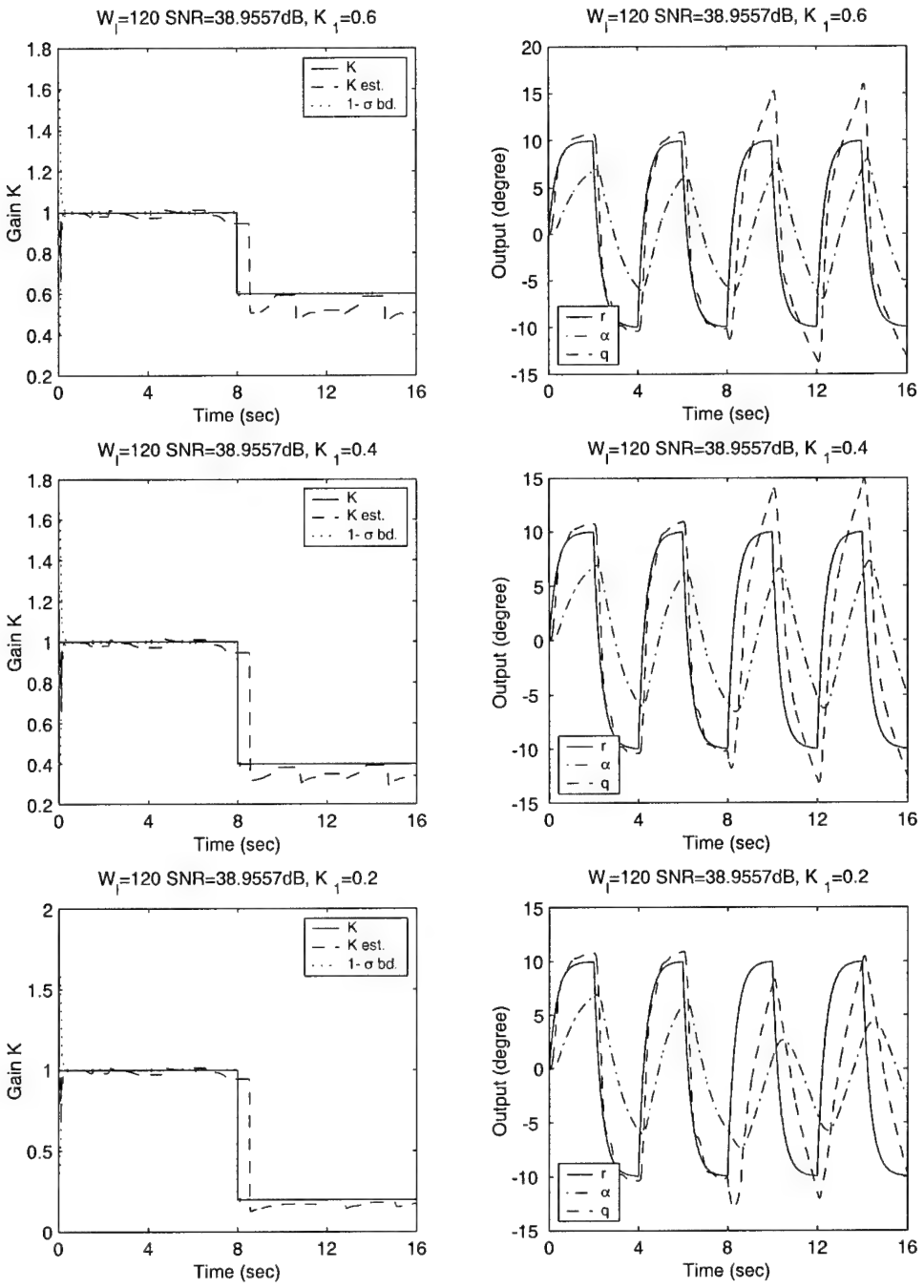


Figure 83. Moving window system ID algorithm ($wl = 120$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

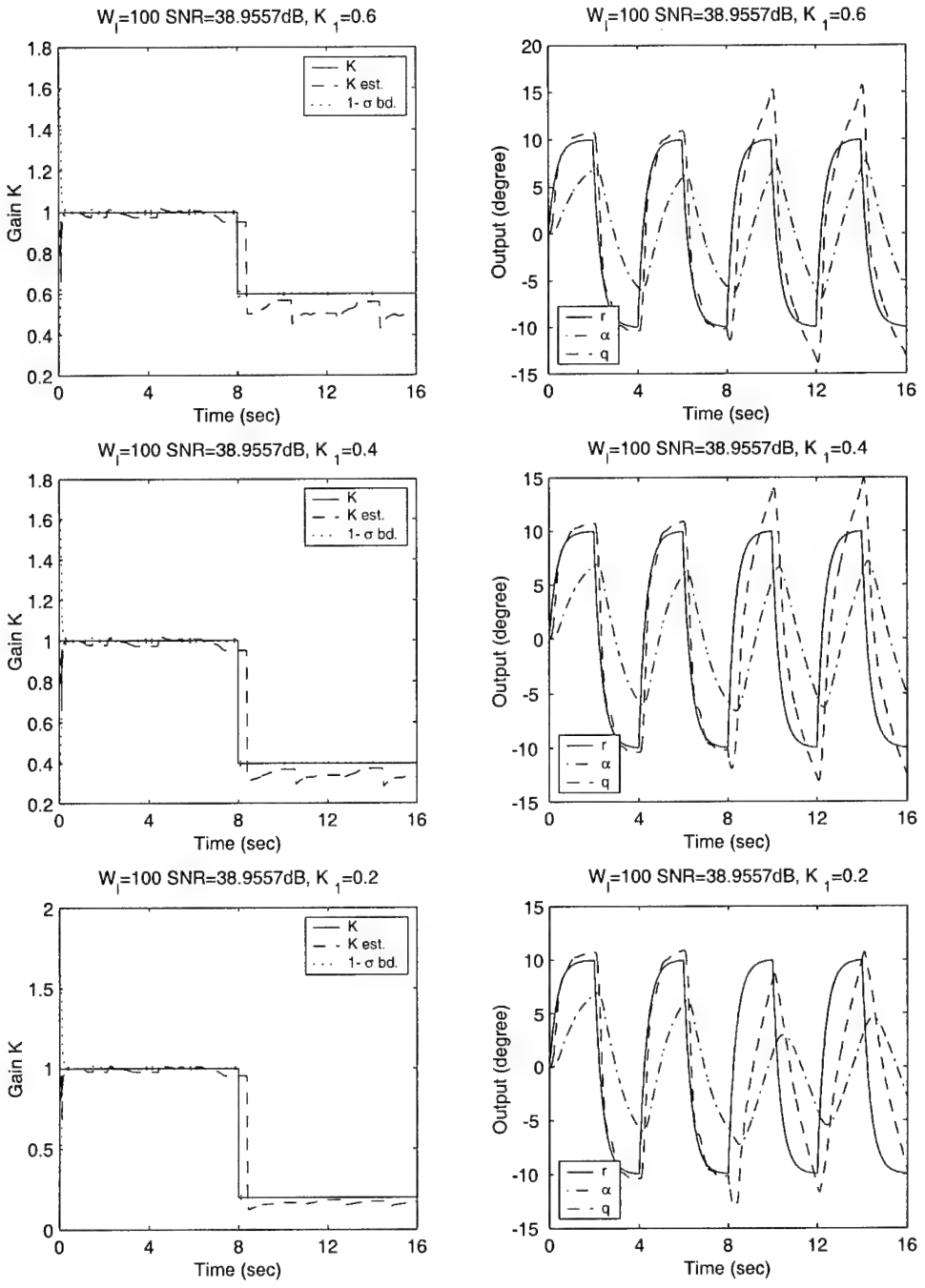


Figure 84. Moving window system ID algorithm ($wl = 100$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

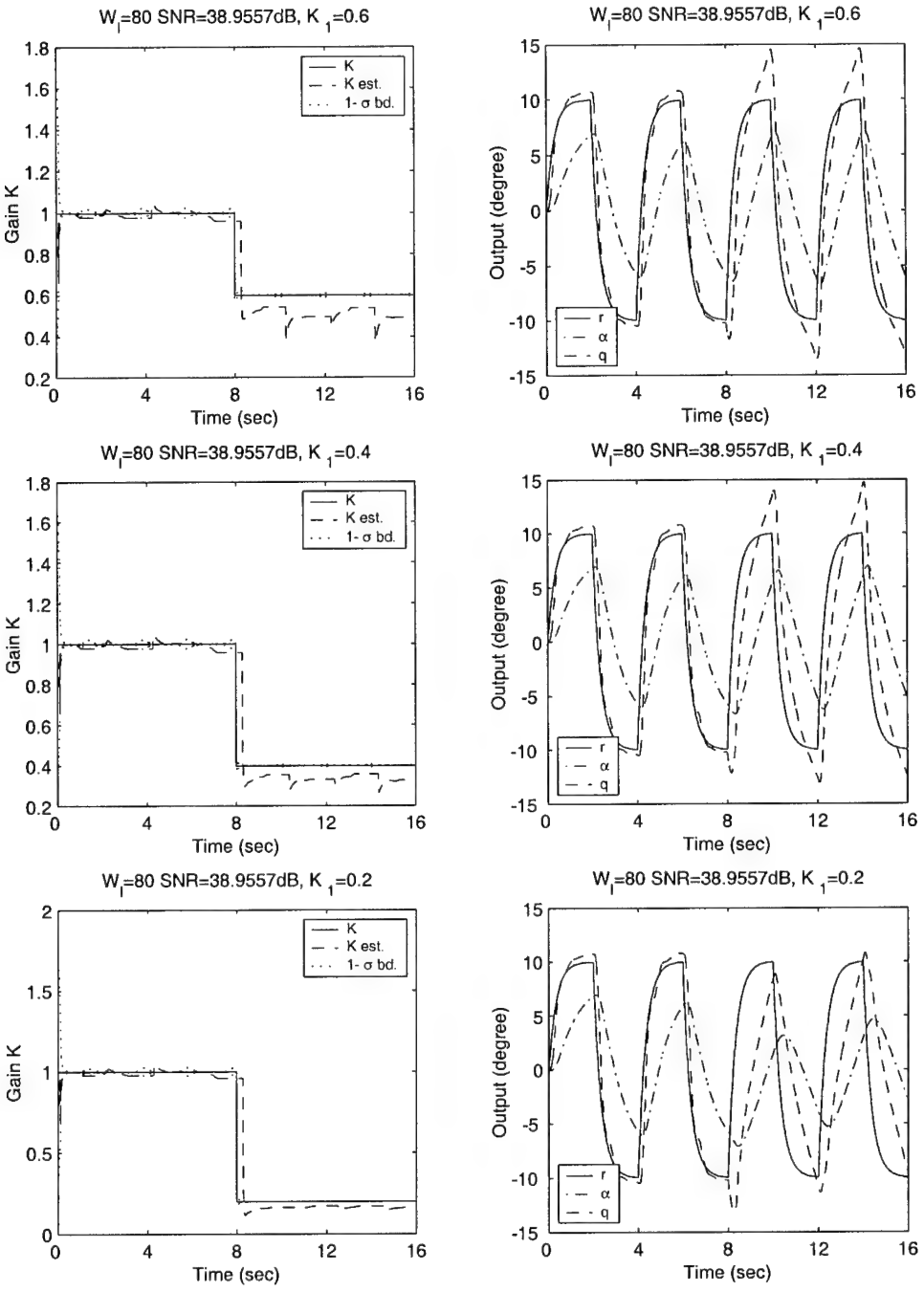


Figure 85. Moving window system ID algorithm ($wl = 80$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

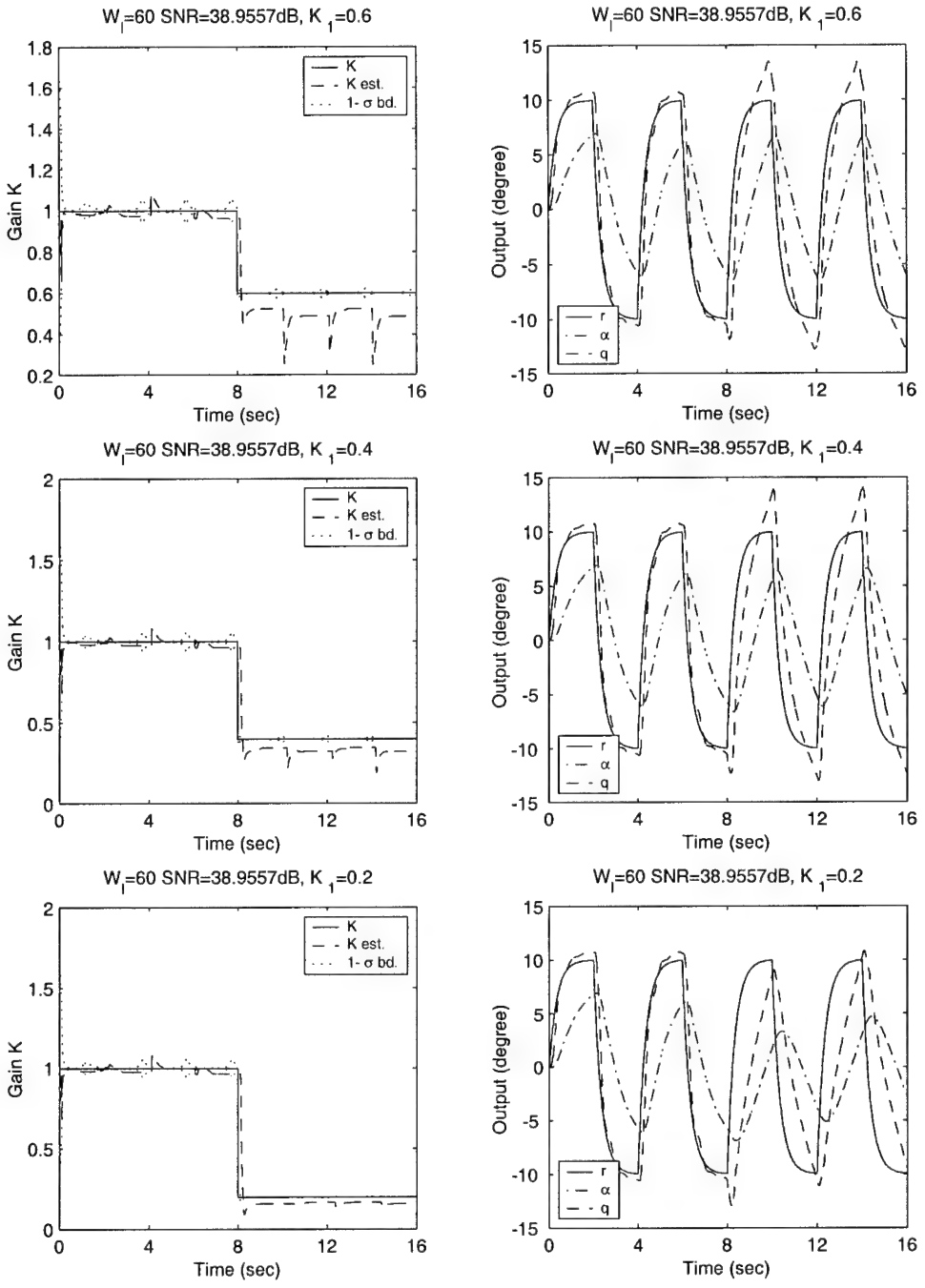


Figure 86. Moving window system ID algorithm ($wl = 60$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

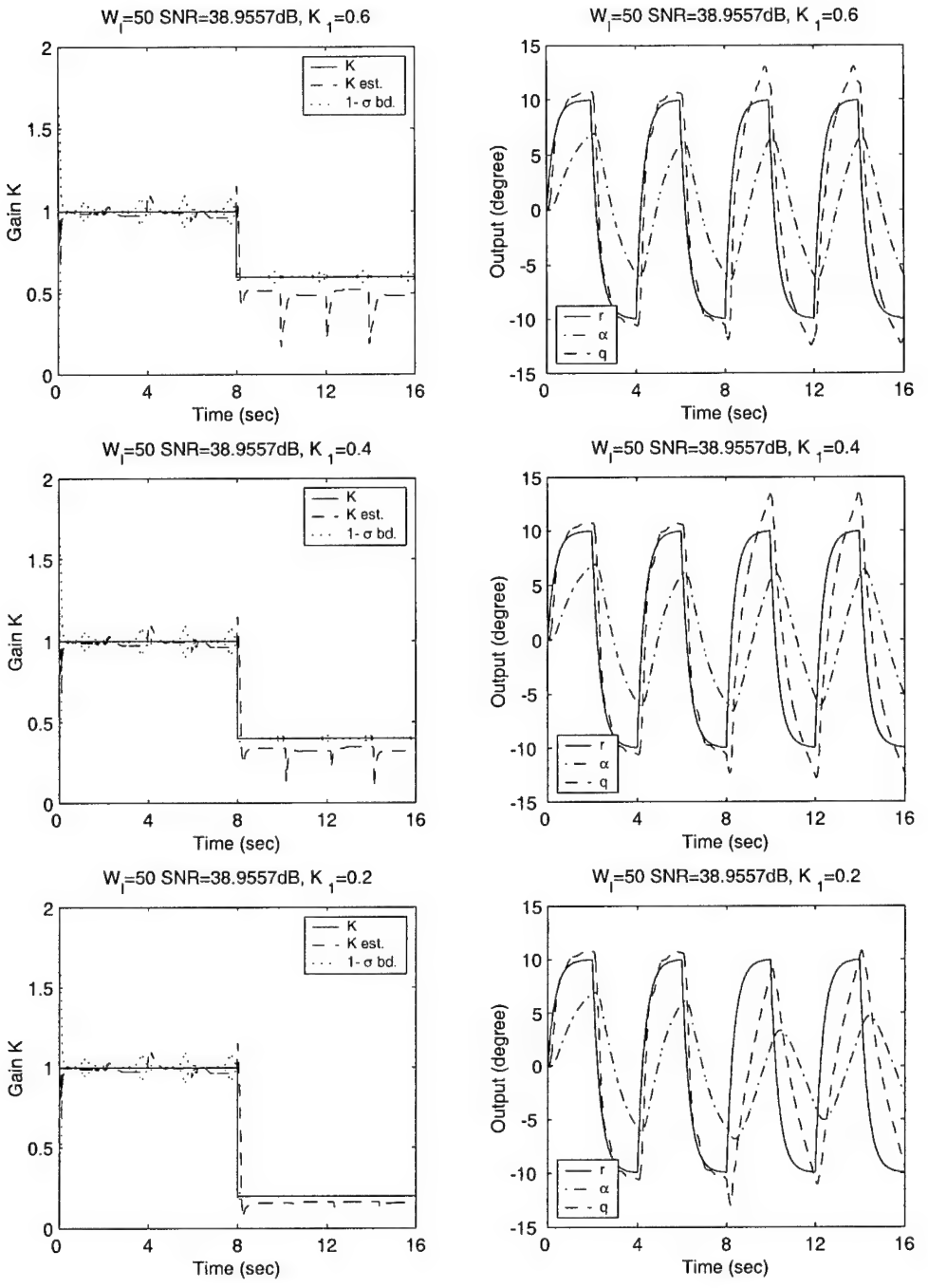


Figure 87. Moving window system ID algorithm ($wl = 50$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

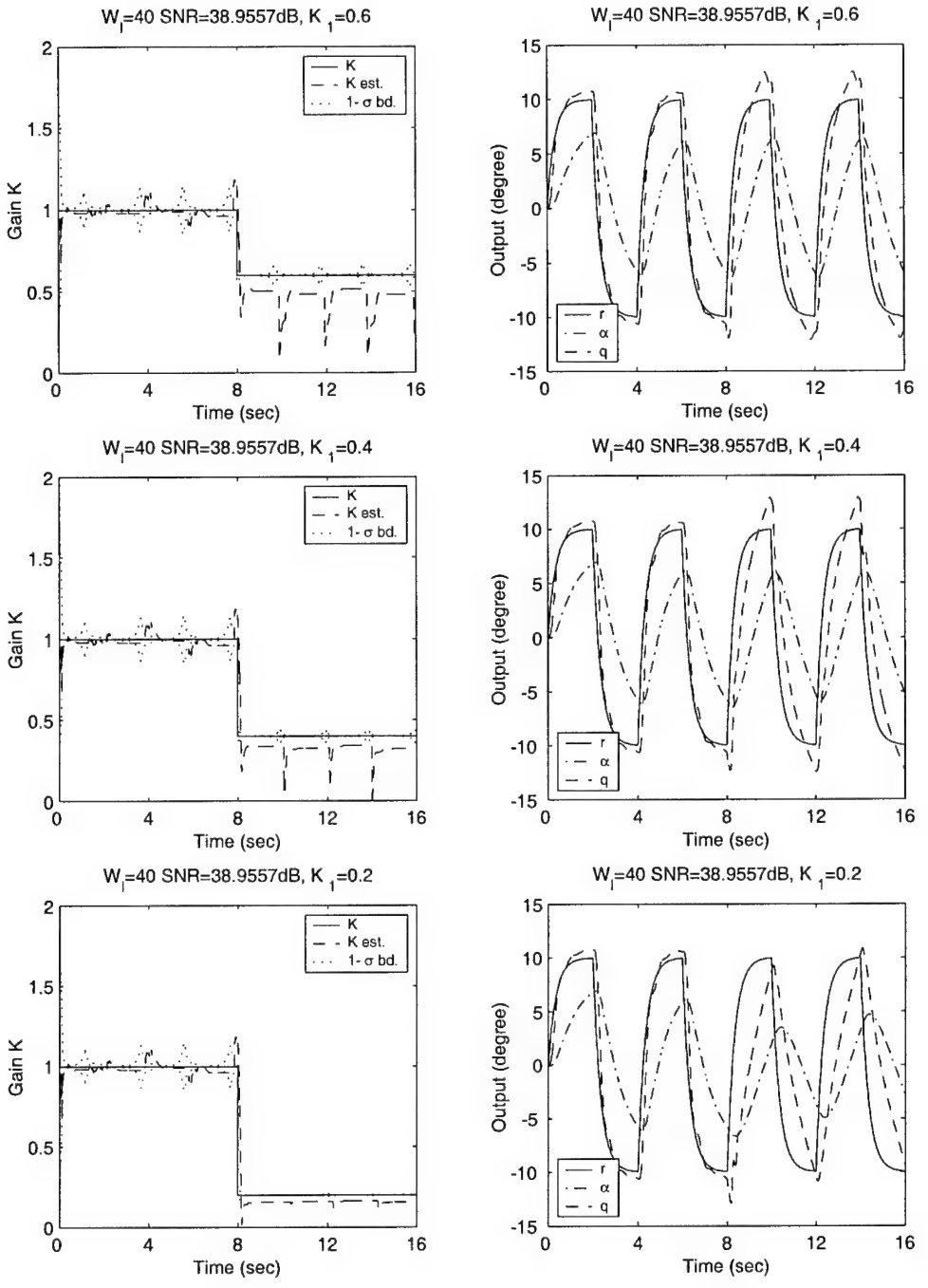


Figure 88. Moving window system ID algorithm ($wl = 40$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

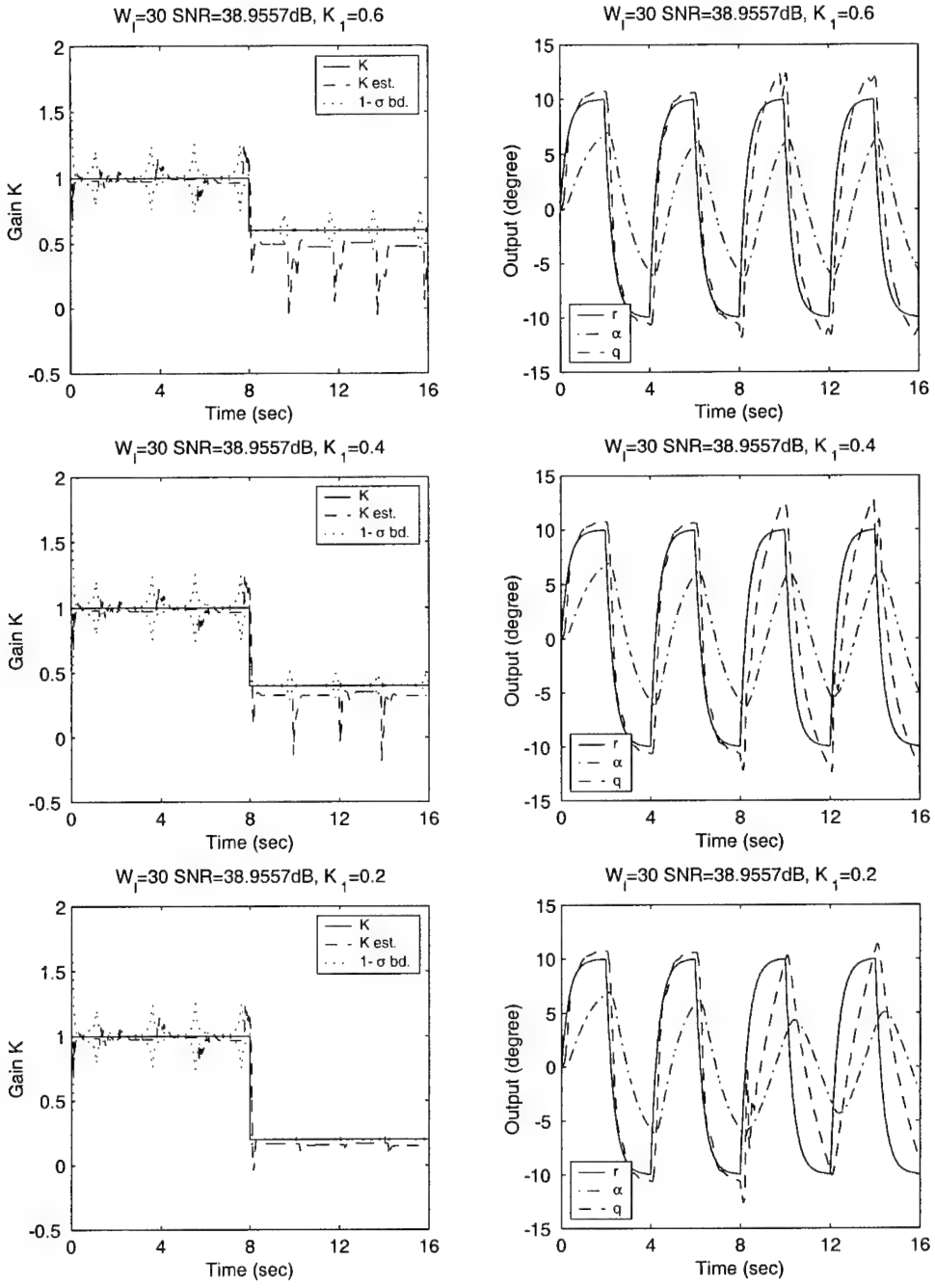


Figure 89. Moving window system ID algorithm ($wl = 30$), with adaptive smoother. Phugoid dynamics, fourth-order actuator model and parametric modeling error ($M_\alpha = 5$ after failure) are included. $K_{\delta_e} = -1.5$. $\sigma_\alpha = 0.03\,\text{deg}$ and $\sigma_q = 0.1108\,\text{deg}/\,\text{sec}$. $K_1 = 0.6$, 0.4 and 0.2.

Table 10. Estimation performance in the presence of Phugoid dynamics, 4th-order actuator and Parameter Modeling Error.

Window Length		Without Modeling Error after Failure				With Modeling Error after Failure			
		Before Failure		After Failure		Before Failure		After Failure	
		$M_\alpha = 3.5$		$M_\alpha = 3.5$		$M_\alpha = 3.5$		$M_\alpha = 5$	
		%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$
120	$K_1 = 0.6$	8.22%	9.8170	0.50%	36.0085	8.22%	9.8156	0.50%	77.6210
120	$K_1 = 0.4$	8.22%	11.5906	0.25%	57.9225	8.22%	11.5887	0.50%	68.3467
120	$K_1 = 0.2$	8.22%	13.8252	0.63%	83.5609	8.22%	13.8230	0.50%	73.8257
100	$K_1 = 0.6$	12.41%	8.0224	0.50%	29.5165	12.41%	8.0159	0.50%	70.6330
100	$K_1 = 0.4$	12.41%	9.1891	0.25%	48.3235	12.41%	9.1811	0.50%	58.0828
100	$K_1 = 0.2$	12.41%	10.5631	0.63%	72.5651	12.41%	10.5541	0.63%	63.5088
80	$K_1 = 0.6$	15.53%	6.3104	0.37%	20.1849	15.53%	6.3061	0.50%	60.7597
80	$K_1 = 0.4$	15.53%	6.9205	0.25%	33.9788	15.53%	6.9153	0.88%	52.6956
80	$K_1 = 0.2$	15.53%	7.6313	0.88%	65.5666	15.53%	7.6254	0.75%	59.2076
60	$K_1 = 0.6$	15.79%	4.1639	0.50%	11.6378	15.79%	4.1623	1.13%	41.1995
60	$K_1 = 0.4$	15.79%	4.3869	0.50%	16.8761	15.79%	4.3848	0.88%	40.7393
60	$K_1 = 0.2$	15.79%	4.6597	0.63%	45.0475	15.79%	4.6572	1.38%	42.5600
50	$K_1 = 0.6$	20.11%	3.3352	0.88%	8.3933	20.11%	3.3340	1.13%	31.4682
50	$K_1 = 0.4$	20.11%	3.4691	0.37%	11.0310	20.11%	3.4677	1.38%	31.5668
50	$K_1 = 0.2$	20.11%	3.6361	0.88%	33.4663	20.11%	3.6343	1.25%	32.6856
40	$K_1 = 0.6$	20.63%	2.4933	2.88%	5.6118	20.63%	2.4923	1.25%	22.1051
40	$K_1 = 0.4$	20.63%	2.5778	1.38%	6.8140	20.63%	2.5765	2.00%	23.1365
40	$K_1 = 0.2$	20.63%	2.6847	1.75%	21.6335	20.63%	2.6832	2.50%	22.2451
30	$K_1 = 0.6$	29.18%	1.7022	6.63%	3.7504	29.18%	1.7013	2.13%	13.6519
30	$K_1 = 0.4$	29.18%	1.7489	4.37%	4.1107	29.18%	1.7476	2.25%	15.2254
30	$K_1 = 0.2$	29.18%	1.8082	4.50%	11.8786	29.18%	1.8066	3.38%	14.6092

Table 11. Estimation performance without Phugoid dynamics, 4th-order actiator and Parameter Modeling Error.

Window Length		Without Unmodeled Dynamics			
		Before Failure		After Failure	
		%	$\sigma_{K_e}/\overline{\sigma}_K$	%	$\sigma_{K_e}/\overline{\sigma}_K$
120	$K_1 = 0.6$	5.43%	7.1464	5.36%	27.1328
120	$K_1 = 0.4$	5.43%	9.9099	5.37%	50.8629
120	$K_1 = 0.2$	5.43%	12.8429	11.75%	78.3058
100	$K_1 = 0.6$	8.99%	4.8353	6.25%	17.2179
100	$K_1 = 0.4$	8.99%	6.4735	4.88%	35.0398
100	$K_1 = 0.2$	8.99%	8.2287	10.50%	56.8502
80	$K_1 = 0.6$	18.72%	3.0250	9.75%	9.7578
80	$K_1 = 0.4$	18.72%	3.9587	7.50%	23.5718
80	$K_1 = 0.2$	18.72%	4.9645	19.00%	49.9900
60	$K_1 = 0.6$	26.45%	1.7476	21.00%	4.7322
60	$K_1 = 0.4$	26.45%	2.2178	19.13%	10.7741
60	$K_1 = 0.2$	26.45%	2.7374	33.37%	36.5729
50	$K_1 = 0.6$	38.62%	1.2928	31.75%	3.0876
50	$K_1 = 0.4$	38.62%	1.6003	30.38%	6.4874
50	$K_1 = 0.2$	38.62%	1.9460	46.00%	27.7534
40	$K_1 = 0.6$	48.49%	0.9132	42.25%	1.9553
40	$K_1 = 0.4$	48.49%	1.0884	41.63%	3.6867
40	$K_1 = 0.2$	48.49%	1.2927	52.12%	18.6363
30	$K_1 = 0.6$	57.33%	0.8237	49.38%	1.5597
30	$K_1 = 0.4$	57.33%	0.8901	52.88%	1.9257
30	$K_1 = 0.2$	57.33%	0.9746	53.63%	10.7669

ing performance. The fourth-order actuator will degrade the tracking performance, but enhance the estimation performance. Using both the Phugoid dynamics and the fourth-order actuator model in our simulation experiment, clearly illustrates the interplay between the various unmodeled dynamics effects on estimation and tracking performance. However, the parametric modeling error not only impairs the estimation performance of the moving window system identification algorithm, but it also degrades the tracking performance.

In the presence of parametric modeling error, the reliability of the system identification algorithm is a major concern. Thus, the estimator's performance is gauged to a large extent by the ratio $\sigma_{K_e}/\overline{\sigma}_K$, where σ_{K_e} is the experimentally recorded parameter estimation error variance, and $\overline{\sigma}_K$ is the average of the predicted parameter estimation error variance. Unfortunately, the latter is close to 1 when the parameter estimation error variance is large. This, in turn, is the motivation for using a parameter estimate smoother before using the parameter estimate to adjust the controller. The latter invariably introduces an estimation lag, which however is minimized when an adaptive parameter estimate smoother is used. With the adaptive smoother, the estimation performance is enhanced, and so is the tracking performance.

Most importantly, the benefit of adaptive and reconfigurable control is amply illustrated in Figure 82. It is apparent that in the case of a sever failure, with a fixed PI controller a departure is on hand, whereas the adaptive and reconfigurable controller yields acceptable tracking performance.

VIII. Conclusion and Recommendations

8.1 Conclusion

An adaptive and reconfigurable flight control system is developed. The novel three-module controller consists of 1) a system identification module, 2) a parameter estimate smoother, and 3) a robust proportional and integral compensator for tracking control.

Two basic system identification algorithms were described that were useful for the determination of the unknown parameters of a dynamical system. We are particularly interested in on-line system identification for adaptive and reconfigurable flight control. We discussed modeling error, disturbances, and the importance of not over-modeling. First, a frequency domain system identification approach for estimating the unknown parameters of an n^{th} order continuous-time SISO system was discussed. Second, an estimation algorithm was also given when measurement noise was taken into account in the modeling of the dynamical system. Because of measurement noise, careful stochastic modeling was used and a modified Least Squares algorithm was developed. It was shown how the effect of the noise can be represented as a weighting matrix, R, in the Least Squares algorithm, and, when applied to the least squares algorithm, it provides accurate parameter estimates.

A new system identification algorithm was developed to identify the plant's control matrix, viz., the plant's open-loop gain, K. We allow for measurement noise, which is injected into the α (angle of attack) and q (pitch rate) channels and propagates through the feedback control system. The system identification algorithm is akin to a Kalman filter and provides estimates of the states, α and q, and the critical open-loop gain plant parameter. The Kalman equations were manipulated so that the loop gain can be estimated and explicit formulae for the loop-gain estimate and the predicted estimation error variance were derived. The rigorous system identification algorithm operates in the presence of measurement

noise and provides an unbiased loop gain parameter estimate and a reliable predicted parameter estimation error variance. On-line operation is achieved, small samples are used, and no human intervention is required. The algorithm is concisely stated in Theorem 1.

High levels of measurement noise and poor excitation increase the parameter estimation error variance and this causes the parameter estimate to fluctuate as we move from window to window. An adaptive parameter estimate smoother reduces the fluctuations automatically in the plant gain estimate used in the "on-line designed" compensator. This improves control performance and reduces bursting, caused by instances of poor excitation. The adaptive parameter estimate smoother uses all the available information on the plant parameter provided by the upstream on-line system identification module and hence the lag, and the error in the plant parameter estimate calculated by the smoother and sent to the compensator, is minimized. Indeed, the role of the parameter estimate smoother is : 1) to reduce the inevitable fluctuations in the parameter estimate prior to using the latter in the downstream on-line controller synthesis algorithm, and 2) to address the ill effects of modeling error and, in particular parametric modeling error, on the performance of the system identification algorithm; the latter are an additional cause of large fluctuations in the parameter estimate as we move from window to window, and to make matters worse, the parameter estimation error prediction is then not reliable.

Moreover, a model based robust PI tracking controller using full state feedback was synthesized. The method used to augment the dynamics to include integral action in a state space formulation was introduced, and the appropriate tracking control law was derived. The robust PI tracking controller provided the performance benchmark against which the performance of our adaptive and reconfigurable controller was gauged.

Extensive simulations were performed to validate the novel adaptive and reconfigurable flight control system. First, the performance of our on-line system iden-

tification algorithm was thoroughly investigated. The algorithm was tested in the presence of the fixed PI controller. The expanding horizon Kalman estimates were compared to the estimates from the moving-window algorithm. The performance of both methods was analyzed and the methods were compared for their ability to identify a failure quickly, viz., a loss of control surface area.

When the fast moving-window system identification was implemented, spots of poorer estimation performance, manifested as spikes in the parameter estimate, were observed at time instants where the pilot's reference signal peaks. To help correct this, a fixed-weights low-pass filter (smoothing module) for the parameter estimate was initially tested for different levels of smoothing action. The fixed-weights parameter estimate filter introduces a lag into the estimation process. To address this problem, an adaptive smoother was developed to reduce the fluctuation automatically in the parameter estimate and the estimation lag, and it was shown to outperform the fixed-weights smoother. Moreover, the bursting phenomenon is automatically mitigated. When the window size is increased, the estimate's fluctuations decreased. However, a shorter window is able to detect a failure faster than a longer window, which is most desirable in reconfigurable control.

The pilot-like 13-bit Barker code pitch rate command sequence excitation was also used in the simulation experiments in a dynamic tracking scenario and the estimation performance of our moving-window system identification algorithm was evaluated. The moving-window system identification algorithm preforms very well. Indeed, the input signal strongly affects the estimation performance of the system identification algorithm - as opposed to classical linear state estimation, i.e., Kalman filtering.

The effects of measurement's Signal to Noise Ratio was investigated. When measurement's SNR increases, the accuracy of the parameter estimate increases, and the tracking performance also increases. The fluctuations in the gain estimate,

and bursting, caused by instances of poor excitation is reduced significantly in high SNR.

The choice of window length for moving-window system identification algorithm deeply affects the performance. Using longer window lengths can reduce the fluctuations in parameter estimate, and bursting, and yield better parameter estimates and better tracking performance.

Attention was also given to unmodeled dynamics effects. Simulations including unmodeled Phugoid dynamics, a fourth-order actuator model, and parametric error, were performed. The unmodeled Phugoid dynamics degrade the estimation performance, and the fourth-order actuator unmodeled dynamics reduce the tracking performance. Including both, gives similar tracking performance as with the fourth-order actuator model only, but better estimation performance than with Phugoid dynamics only. However, parametric modeling error significantly degrades the estimation and tracking performance of the moving-window system identification algorithm and is the driving force for using a parameter estimate smoothing module.

In summary, in the novel three-module adaptive and reconfigurable controller, the reciprocal of the estimated loop gain derived from the system identification algorithm and processed by the smoothing module is used on-line to adjust the compensator, to account for the failure, and thus recover performance. The tracking performance of the complete adaptive and reconfigurable control system is shown to be superior to the tracking performance of the robust, but fixed, PI tracking controller, in particular, in the case of a severe failure.

The adaptive and reconfigurable controller design methodology developed in this dissertation is illustrated in a flight control context. However, this development is applicable to a broad range of control problems.

8.2 Contributions

1. Developed three-module adaptive and reconfigurable controller which includes an adaptive parameter estimate smoother. Improves estimation performance and combats bursting phenomenon.
2. Established the parameters for the optimal operation of the system identification module, viz., the sampling rate and the window length, and experimentally investigated the SNR effect.
3. Gave attention to the design of a robust tracking controller which accommodates a control surface loss.
4. Developed remedial action for the accommodation of modeling errors in indirect adaptive control; this includes low frequency unmodeled dynamics, high frequency unmodeled dynamics, and parametric modeling error.

8.3 Recommendations for Future Research

1. This dissertation presented only the single input (pitch rate command) of the F-16 class aircraft. The three-module adaptive and reconfigurable controller developed in this dissertation should be applied to multiple-input signals to test it's ability. One can also create a scenario where the B matrix is changed due to structural damage.
2. To simplify the development of the three-module adaptive and reconfigurable controller, this dissertation deal with the measurement's noise only. Process noise may be included in the model to give a great insight on how this controller is performing.
3. The system identification module developed in this dissertation to estimate the states and the open-loop gain only. The results of the experiments indicated that the parameter modeling error (M_α) indeed affected the performance of the system identification module. We mitigated the bad effects by using the

adaptive smoother and modifying the control gain (K_{δ_e}). One could develop another smoothing filter and fine tune the PI controller to improve the performance of the system identification module when more modeling errors are included. One could even modify the system identification module to also estimate the stability derivatives.

Bibliography

1. Bar-Shalom, Y., and Tse, E., "Dual Effect, Certainty Equivalence and Separation in Stochastic Control," IEEE Trans. Automatic Control Vol. AC-19, 1974, pp. 494-500.
2. Brown, J. M, II., "Optimal Inputs for System Identification," PhD dissertation, AFIT/DSG/ENG/95-S-04, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, September 1995.
3. Caglayan, A. K., et. al., "Detection, Identification and Estimation of Surface Damage/Actuator Failure for High Performance Aircraft," Proceedings of the 7th American Control Conference. Atlanta, Georgia, June 1988, pp. 2206-2212.
4. Chandler, P. and Pachter, M., "On-Line Optimizing Networks for Reconfigurable Control," Proceeding of the 32nd Conference on Decision and Control, San Antonio, Texas, December 1993, pp. 2272-2277.
5. Chandler, P. and Pachter, M., "Constrained Linear Regression for Flight Control System Failure Identification." Proceedings of the American Control Conference, San Francisco, California, June 1993, pp. 3141-3145.
6. Chandler, P. and Pachter, M., "Regression Techniques for Aircraft Parameter Identification from Noisy Measurements in Maneuvering Flight," Proceedings of the 31st Conference on Decision and Control, Tucson, Arizona, December 1992, pp. 2311-2316.
7. Chandler, P., Pachter, M., and Mears, M., "System Identification for Adaptive and Reconfigurable Control," Journal of Guidance, Control and Dynamics, Vol. 18, No.3, June 1995, pp. 516-524.
8. Chen, Han-Eu., "Recursive Estimation and Control for Stochastic Systems," New York: John Wiley and Sons, Inc., 1985.
9. D'Azzo, John J. and Houpis, Constantine H., "Linear Control Systems Analysis and Design: Conventional and Modern (4th edition)," New York: McGraw-Hill, Inc., 1995.
10. Franklin, Gene F, "Digital Control of Dynamic Systems (3rd edition)," Menlo Park California: Addison Wesley Longman, Inc., 1998.
11. Houpis, C. H. and Meir Pachter. "Application of QFT to Control System Design - An Outline for Engineers" International Journal of Robust and Nonlinear Control, Vol. 7, No 6, June 1997, pp. 515-531.
12. Lainiotis, D. G., Deshpande, J. G., and Upadhyay, T. N., "Optimal Adaptive Control : A Nonlinear Separation Theorem," Internat. J. Control, Vol. 15, 1972, pp. 877-888.

13. Ljung, L., "System Identification for the User." New Jersey: Prentice Hall, 1987.

14. The MathWork, Inc., 21 Elliot Street, Natick, MA 01760. MATLAB. September 1999. Version 5.3.1.

15. Maybeck, P., "Stochastic Models, Estimation and Control. Volume 1," Arlington, VA: Navtech Book and Software Store, 1994.

16. Maybeck, P., "Stochastic Models, Estimation and Control, Volume 2," Arlington, VA: Navtech Book and Software Store, 1994.

17. Maybeck, P., "Stochastic Models, Estimation and Control, Volume 3," New York, NY: Academic Press, 1982.

18. Nelson, R., "Flight Stability and Automatic Control," New York, NY: WCB/McGraw-Hill, Inc., 1998.

19. Norton, J. P., "An Introduction to Identification," London: Academic Press, 1986.

20. Olmstead, D. N., "Optimal Feedback Controls for Parameter Identification," PhD dissertation, AFIT/DS/EE/78-5, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, March 1979.

21. Pachter, M., Class Notes from EENG 640: Flight Dynamics I. Air Force Institute of Technology, Wright-Patterson AFB, OH, 1998.

22. Pachter, M., Class Notes from EENG 708: Multivariable Systems. Air Force Institute of Technology, Wright-Patterson AFB, OH, 1998.

23. Pachter, M. and Chandler, P., "Control Reconfiguration with Actuator Rate Saturation," Proceedings of the 14th American Control Conference, Seattle Washington, June 1995, pp. 3495-3499.

24. Pachter, M. and Chandler, P., "Reconfigurable Tracking Control with Saturation," Journal of Guidance, Control and Dynamics, Vol. 18, No. 5, Sept-Oct 1995, pp. 1016-1022.

25. Schmotzen, R. E., and Blankenship, G. L., "A Simple Proof of the Separation Theorem for Linear Stochastic System with Time Delay," IEEE Trans. Automatic Control, Vol. AC-23, 1978, pp. 734-735.

26. Sillence, J. P., "Loop Gain Estimation for adaptive Control," Master thesis, AFIT/GE/ENG/99M-27, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, March 1999.

27. Sworder, S. C., and Sworder, D. D., "Feedback Estimation Systems and the Separation Principle of Stochastic Control," IEEE Trans., Automatic Control, Vol. AC-16, 1971, pp. 350-354.

Vita

Lt. Col. Yih-Shiun Huang was born 12 February 1965 in Taoyuan, Taiwan, The Republic of China. He graduated from Chung-Cheng High School (Taipei) in 1983 and attended Chung-Cheng Institute of Technology in Taoyuan. In 1987, Lt. Col. Huang graduated from Chung-Cheng Institute of Technology with a Bachelor of Electrical Engineering. His first assignment was to the 10th Field Army as a platoon leader in Taichung. In 1990, he attended Chung-Cheng Institute of Technology and graduated with a Master of Electronic Engineering after two years of research. He was assigned to the Weapon System Acquisition Management Office in Headquarters of the Republic of China Air Force as an Avionics System Engineer in 1992. In 1996, Lt. Col. Yih-Shiun Huang was competitively selected by the Republic of China Air Force to attended the Graduate School of Engineering and Management, Air Force Institute of Technology (AFIT) to pursue a Ph.D. in Electrical Engineering. Lt. Col. Yih-Shiun Huang is a member of Tau Beta Pi honor society.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
06-03-2001	Ph. D. Dissertation	Sep 1999 – Mar 2001

4. TITLE AND SUBTITLE

ADAPTIVE AND RECONFIGURABLE FLIGHT CONTROL

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Huang, Yih-Shiun, Lt. Col., ROCAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 P Street, Building 640
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/DS/ENG/01-02

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Philip Chandler
AFRL/VACA
210 8th St. BLDG. 146
Wright-Patterson AFB, OH 45433-77652
(937) 255-8680

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

An indirect adaptive and reconfigurable flight control system is developed. The three-module controller consists of 1) a system identification module, 2) a parameter estimate smoother, and 3) a proportional and integral compensator for tracking control. Specifically: 1) The identification of a linear discrete-time control system's open-loop gain is addressed. The classical Kalman filter theory for linear control systems is extended and the control system's state and loop gain are jointly estimated on-line. Explicit formulae for the loop gain's estimate and estimation error covariance are derived. The estimate is unbiased and the predicted covariance is reliable. 2) An adaptive smoother is developed to reduce the fluctuations automatically in the gain estimate, and bursting, caused by instances of poor excitation. 3) Special attention is given to the design of a proportional and integral tracking controller. The outputs of the system identification and gain smoother modules are used to adjust the tracking controller's gain continuously in order to compensate for a possible reduction in the loop gain due to control surface area loss, thus achieving the benefits of adaptive and reconfigurable control. The performance of the adaptive and reconfigurable controller in the face of a simulated control surface failure is examined in carefully designed experiments. The adaptive controller developed in this dissertation and illustrated in a flight control context is applicable to a wide range of control problems.

15. SUBJECT TERMS

Three-Module Adaptive and Reconfigurable Flight Control, System Identification, Adaptive Smoothing, PI Tracking Controller, Kalman Filter, Expanding Horizon System Identification, Moving Window System Identification

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr M. Pachter, ENG
U	U	U	UU	167	19b. TELEPHONE NUMBER *(Include area code)* (937) 255-3636, ext 4593

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

Form Approved
OMB No. 074-0188